Tampere University

Feranmi Akanni

# APPLICATION OF MACHINE LEARNING METHODS ON PREDICTIVE MAINTENANCE

Faculty of Information Technology and Communication Sciences(ITC)

Master's thesis

October 2019

# ABSTRACT

Feranmi Akanni: Application of Machine Learning Methods on Predictive Maintenance
Master's thesis, 63 pages
Tampere University
Master's Degree Programme in Computational Big Data Analytics
October 2019

---

Maintenance is important in optimization of the business value of a functional unit and this optimization can only be achieved through predictive maintenance which ensures that maintenance activities are not carried out before due time and at the same time, prevent occurrence of breakdown of functional units because of missed maintenance activities.

In this thesis, we focus on using different machine learning methods to predict the failure of a functional units. We explore the data and use missing data techniques to deal with missing values in the dataset, which resulted in a complete dataset. We explore various feature selection techniques to extract important features and reduce dimensionality of the dataset. Then, we explore the following machine learning methods: logistic regression, naïve Bayes, support vector machine, k-nearest neighbour and ensemble learning techniques which are bagging and boosting methods. Our results indicated that predictions from ensemble learning techniques have better evaluation metrics compared to other machine learning methods.

Keywords: predictive maintenance, machine learning, missing data, features selection techniques, classification, evaluation metric.

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to all my teachers in University of Tampere, most especially to my course adviser and my supervisor for this project, Professor Martti Juhola for his knowledge impact ability, support, and guidance during my years of study in University of Tampere and toward successful completion of this project. I would also like to appreciate SCANIA AB organization for making their data available in public domain for experimental purpose.

My appreciation goes to my family members and friends for their unending support, contribution and understanding during my years of study in Tampere. Finally, I thank almighty God for seeing me through years of my study.

Tampere, 2 October 2019

Feranmi Akanni

# Table of Contents

# Chapter One

The main objective of this study is the application of different machine learning methods in predictive maintenance. This section contains an introduction to maintenance and machine learning, and a literature review of related work for this study.

## 1.1 Introduction

Maintenance can be described as the set of activities and actions which involve functional checking, servicing, testing, measurement, repairing or replacing of devices, equipment, machineries, and supporting utilities in industrial, business, governmental and residential environment [1]. Maintenance can also be defined as the combination of all technical and associated administrative actions intended to retain an item in or restore it to a state in which it can perform its required function (British standard glossary of terms used in terotechnology, 1993) [2].

Maintenance is important in ensuring that the functional units are effective in their performance, in preserving the life span of the functional unit and in contributing to the sustainability and availability of the functional units. The lack or ineffectiveness of maintenance practices can contribute negative effects to the overall business performance through their impact on quality, the availability of the equipment, the organization competitiveness and the organization environment.

There are three main types of maintenance, which are corrective, preventive and predictive maintenance. Corrective maintenance is a type of maintenance where maintenance activities are carried out after the breakdown or malfunctioning of the equipment. Preventive maintenance is also referred to as predetermined preventive maintenance and is a type of maintenance where maintenance activities are carried out on the equipment at fixed interval to avoid malfunctioning or breakdown of the equipment. These two types of maintenance are referred to as traditional maintenance strategies. Predictive maintenance is also referred to as condition-based maintenance (CBM). CBM is a set of maintenance actions based on the real-time or near real-time assessment of equipment condition, which is obtained from embedded sensors and/or external tests and measurements, taken by portable equipment and/or subjective

condition monitoring [4]. Predictive maintenance is maintenance carried out following a forecast derived from repeated analysis or known characteristics and evaluation of the significant parameters of the degradation of the equipment [5].

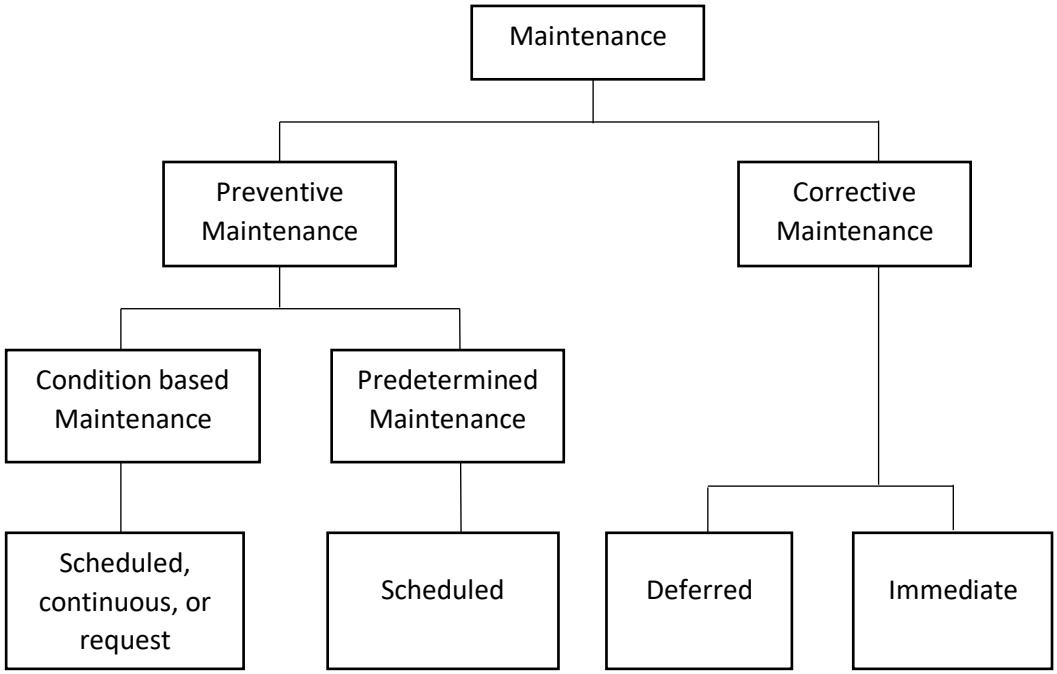


Figure 1: Overview of the different maintenance types (BS-EN 13306, 2010, p.20)

The advancement in technologies has significantly contributed to the evolution of maintenance activities over the past decades. Jantunen et al. [6] suggest that the concept of maintenance has evolved over the last few decades from a corrective approach (maintenance actions after a failure) to a preventive approach (maintenance actions to prevent the failure). Notably, the path of evolution of the maintenance activities has been from non-issue to business strategic concern. Initially, maintenance was majorly seen as an inevitable part of production where the maintenance activities were carried out after the breakdown of the equipment because downtime was not a critical issue and it was adequate to carry out maintenance after breakdown.

Later, it was conceived that maintenance was a technical matter and this did not only include optimizing technical maintenance solutions, but it also included the attention of the organization on the maintenance work [7]. Going forward, maintenance was separated from being a subfunction of production and was considered as a functional unit which represents one of the profit contributors to the organization. At this stage,

the downtime from equipment breakdown was a critical issue and maintenance activities were carried out to prevent equipment breakdown.

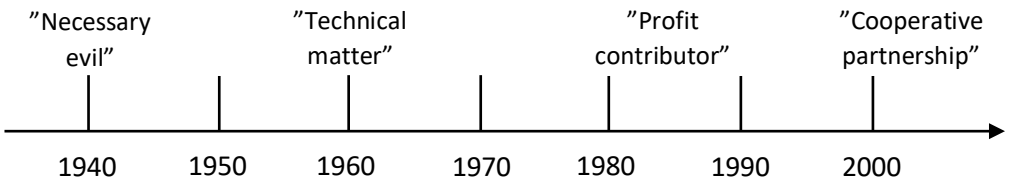


Figure 2: The maintenance function in a time perspective [7].

The major impact of technology advancement in the area of maintenance can be observed in predictive (condition-based) maintenance where sensors are used to measure relatively huge amounts of data about the conditions of the equipment and this data is used to create models using different methods such as machine learning methods to determine the optimal time to carry out maintenance activities on the equipment just before the equipment failure or breakdown. The new technology such as IoT promotes the instantaneous availability and accessibility of the data about the conditions of the machines or products.

Learning is defined according to T. Mitchell [8]; "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E". Machine learning is a specific subfield of Artificial Intelligence that can identify patterns and learn from data through self-learning algorithms to predict the output of future observations. Based on the nature of the business needs to solve and type of data available for analysis, machine learning can be divided in the following categories; supervised learning, unsupervised learning, reinforcement learning.

Supervised learning uses labeled training data to build models which are used to make prediction for future observations. A simplified common application of supervised learning is in a spam email filtering system which contains as training data, labeled emails which are correctly marked spam or not-spam, and this is used to predict the class of new email. Supervised learning can be categorized as regression when the labeled feature is continuous or as classification when the labeled feature is discrete class labels.

Unsupervised learning explores unlabeled data to extract meaningful information from the data. A common method of unsupervised learning is clustering which is an exploratory technique that organizes data objects into meaningful subgroups without prior information about the subgroups in the data objects and this is achieved by grouping data objects that are similar together but are more dissimilar to other data objects in other clusters. Unsupervised learning is carried out on the unlabeled data and this leads to creation of a labeled feature that adds labels to the data. The resulting data from unsupervised learning can be passed on to a supervised learning process.

Reinforcement learning develops a system (agent) that improves its performance based on the interactions with the environment [9]. A simplified common applications of reinforcement learning is in robotics or a chess playing game. In a chess playing game, the agent decides upon a series of moves depending on the state of the chess board which is the environment, and the reward can be defined as win or lose at the end of the game [9]. As the agent interacts with the environment, it uses reinforcement learning to learn a series of actions that maximizes its reward through an exploratory trial-and-error approach or deliberative planning. Figure 3 below is a schematic representation of reinforcement learning;

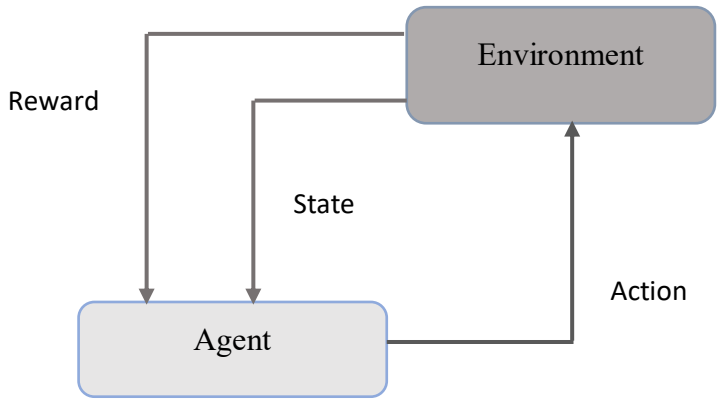


Figure 3: Representation of Reinforcement Learning [9]

Deep learning is a specific method of machine learning that incorporates neural networks or other structures in successive layers to learn from data in an iterative manner and it uses hierarchical neural networks to learn from a combination of unsupervised and supervised algorithms [10]. A common application of deep learning is in image recognition, voice recognition and computer vision. A neural network

involves three or more layers which are the input layer, hidden layer and output layer. The input layer receives the data and various activation amounts for the data are computed at the hidden layers based on the weights of the nodes.

This thesis focuses on applying different classification methods of supervised learning on data collected from a heavy Scania truck. Chapter two involves the description of the data and data source, the description of the data variables and the exploratory analysis of the data, which involves methods of handling missing data, and application of dimensionality reduction techniques for features selection and transformation. Chapter three involves metrics for evaluating performance of a model, and application of different machine learning methods on the data. Chapter four involves application of more different machine learning methods on the data, and the result and comparison of the different machine learning methods. Chapter five presents a summary of entire work and future recommendation are highlighted in the conclusion.

## 1.2 Literature Review

The concept of Predictive Maintenance was introduced by Rio Grande Railway Steel Company in the late 1940s. The company used CBM techniques to monitor critical parameters such as oil and fuel in the engine through changes in temperatures and pressures readings [12]. There are a good number of works on Predictive Maintenance and this literature can be categorised using different criteria such as the source type of the used data, the methods and algorithms for the analysis of data, frameworks or approach.

The review of previous research articles related to predictive maintenance was carried out within time span of year 2001 to 2017, because of the numerous research articles in the field of Predictive maintenance. Marzio, Enrico & Luca [13] used Genetic Algorithm and Monte Carlo (MC) simulation for determining the optimal degradation level at which predictive maintenance must be carried out. S.K. Yang [14] presented the experimental results of predictive maintenance by using Kalman filter. D. Bansal, D. Evans & B. Jones [15] used a neural network approach for real-time predictive maintenance for machine systems. W. Wang [16] used a probabilistic approach to predict both the initiation point of the failure delay period and the remaining life of

production equipment based on condition monitoring information. Stefano, Roberto & Sergio [17] used a neural network method for predictive maintenance of textile machine systems.

The research study of W. Wang & W. Zhang [18] was based on prediction of remaining useful life of an asset using expert judgements based on measured condition monitoring parameters, and stochastic filtering theory was used to predict the remaining useful life given, among other condition monitoring parameters, the available past expert judgments on the same asset to date. Y. G. Li [19] used combined regression methods of both linear and quadratic models to predict the remaining useful life of gas turbine engines. A. Kadir, Sharifah & Takashi [20] used artificial neural networks to predict the remaining useful life of rotary machinery (bearing) for predictive maintenance. Y. Peng & M. Dong [21] used an age-dependent hidden semi-Markov model to predict the equipment health of hydraulic pumps. The CBM is based on the failure rate which is a function of both the equipment age and the monitored conditions of the equipment.

The research study of A. Widodo & B. S. Yang [22] was based on prediction of remaining useful life bearing using combination of both survival probability and support vector machine techniques. The study exploited censored and uncensored data generated through equipment condition monitoring and the survival analysis was carried out on the data to predict the failure rate of the equipment before support vector machine was used as the classifier method. J. Hu, L. Zhang & W. Liang [23] used by dynamic Bayesian network method for predictive maintenance. H. Kim et al. [24] used a support vector machine classifier method for predictive maintenance of bearings of High Pressure-Liquefied Natural Gas (HP-LNG) pumps. J. Yuan & X. Liu [25] used a combination of manifold regularization based semi-supervised learning and dimensionality reduction techniques to perform condition monitoring (CM) for faults diagnosis and prognosis.

The research study of T. Praveenkumar et al. [26] was based on prediction of failure in automobile gearbox using support vector machine on the extracted features from gearbox vibration measurements. M. Zaidan et al. [27] used Bayesian hierarchical models to carry out probabilistic prediction of remaining useful life for aerospace gas turbine engine. Hui & Jianchao [28] predicted the remaining useful life of components that have stochastic dependency using stochastic filtering theory. Riccardo et al. [29]

exploited three classification models which were decision trees, random forests and neural network to a complex high-speed packing machine for making decision related to predictive maintenance and the result of their study revealed that random forest classifier performed better than other two classifier in terms of accuracies of the models. C. Gondek, D. Hafner & O. Sampson [30] used combination of feature engineering and one classification method which was random forest to predict the failure of Air Pressure System of Scania Trucks.

The review of previous works showed that Bayesian based methods and support vector machine have started gaining popularity. Riccardo et al. [29] work focused on three main classifiers which were decision trees, random forests and neural networks. However, the work in this thesis is different from reviewed studies and most especially the last study reviewed because it introduces dimensionality reduction techniques such as features extraction and explores gradient boosting methods. This thesis work used data from the air pressure system (APS) of heavy Scania truck.

# Chapter Two

## 2.1 Data Sources and Description

The dataset for this work was discovered from UCI machine learning repository website https://archive.ics.uci.edu/ml/datasets/APS+Failure+at+Scania+Trucks. In the year 2016, the dataset was donated by the representatives of Scania AB which is a major Swedish manufacturer of commercial vehicles, most especially heavy trucks and buses. Scania AB is also a manufacturer of diesel engines for heavy vehicles, marine and generally for industrial applications. In the same year, the dataset was used for industrial challenge at the international symposium of the intelligent data analysis.

The dataset consists of data generated from everyday utilization of a heavy Scania truck and the main component of focus is Air Pressure System (APS) which generates pressurized air for effective operation of various components of the truck such as brake and gear components. The dataset consists of one response variable which is named class and 170 independent variables which have been anonymized for security purpose and to reduce the risk of unintended usages of the dataset.

The dataset includes the training set which consists of 60,000 instances and the test set which consists of 16,000 instances. The class label of response variable for the training set consists 1,000 positive class and 59,000 negative class while the class label for the test set consists of 375 positive class and 15, 625 negative class. The positive class of the dataset indicates a truck with failures which are related to APS and the negative class of the dataset indicates a truck with failures which are not related to APS. The focus of this work is to achieve minimum type I and type II errors. Type I error refers to false positive which can lead to unnecessary checks and maintenance of a truck. While Type II error refers to false negative which can lead to missing out on a faulty truck that requires maintenance and this can cause breakdown.

The missing data imputation and the exploratory analysis were carried out using R programming and the other preprocessing activities and the models building were carried out using R programming.

## 2.2 Exploratory Data Analysis

Exploratory analysis involves the process of investigating the main characteristics of the dataset and summarize the findings through graphical representations. The exploratory analysis is carried out on the dataset set which is skewed towards negative class which represents 0. Figure 4 represents the skewness of the dataset.

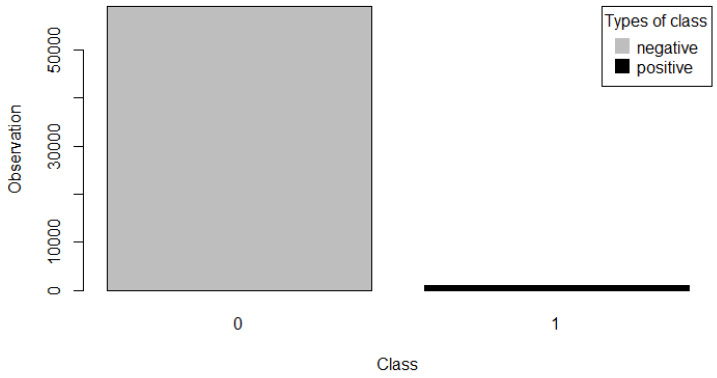


Figure 4: A plot showing the histogram of the target feature (number of observations of each target value)

### 2.2.1 Missing Data Handling

There are missing values in the training set and out of 60,000 cases, there are 591 cases without missing values, and this indicates that deletion of cases with missing values is not suitable for this dataset. Out of 170 independent features, only one feature is without missing values and figure 5 represents the missing value percentage in each feature.

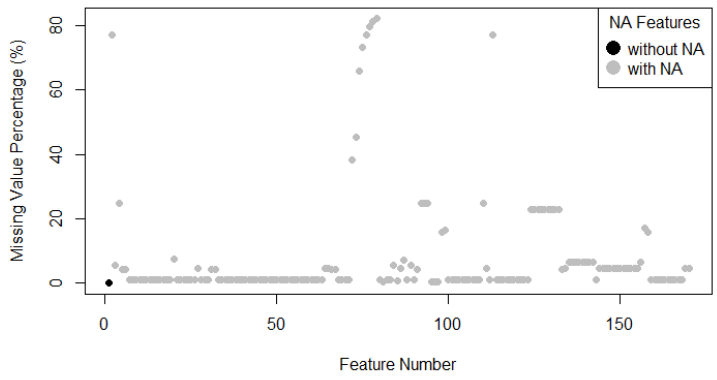


Figure 5: A plot showing percentage of NA (missing value) in each feature before Imputation

Missing at random (MAR) is one of the types of missing data mechanism and data is missing at random when the probability of the missing data on a feature Y depends on the other observed feature(s), but not to the value of Y that should have been observed [37]. The MICE (Multivariate Imputation via Chain Equations) package in R is used to perform missing value imputation and MICE assumes that the data are missing at random (MAR) [36]. The method of MICE was set to classification and regression tree (cart) and figure 6 below represents the missing value percentage in each feature after imputation of missing values were carried out.

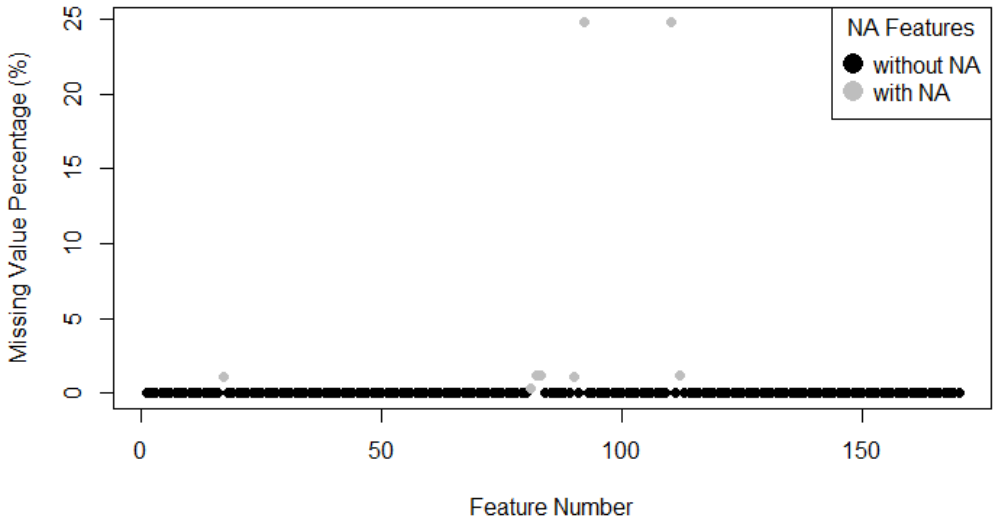


Figure 6: A plot showing percentage of NA in each feature after Imputation

Figure 6 shows that after missing values imputation, the following 8 features are still having missing values;

Features: ah_000, bt_000, bu_000, bv_000, cd_000, cf_000, co_000, cq_000

At this point, the cases with missing values are removed for models that cannot be executed with missing values. After deletion, the complete cases are 44, 667 out of total cases of 60,000 and for models that can be executed with missing values in the dataset, the training with the total cases of 60,000 is used to train the models. Figure 7 below represents that there are no missing values in the version of the dataset of this project work where cases with missing values have been removed.

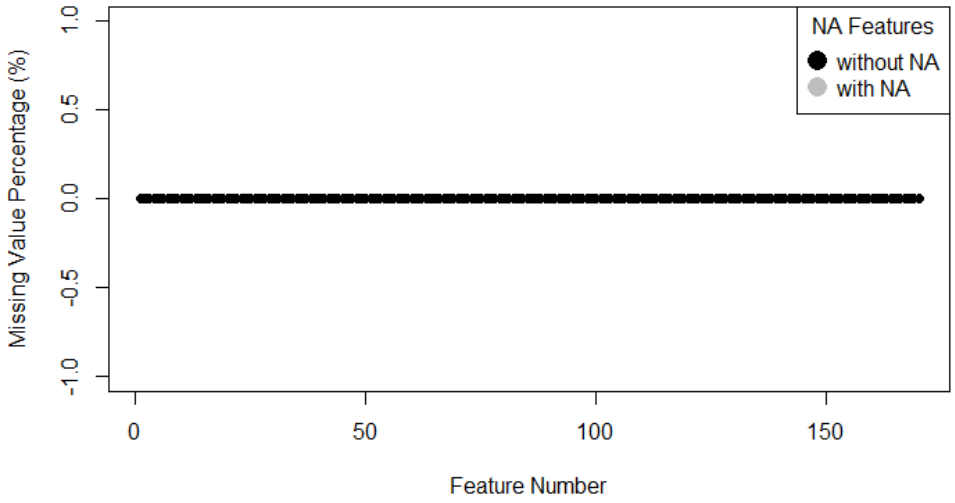


Figure 7: A plot with no NA in the dataset

### 2.2.2 Feature Selection

Feature selection involves choosing a k-dimensional important and relevant feature subspace from initial d-dimensional feature space by picking k of the original features where k is less than d (k<d) and ignoring the remaining (d-k) features which are assumed to be irrelevant features or too noisy to benefit the performance of the models. It is important to carry out feature selection for the following reasons; for improving the performance of predictors in the models, for providing computationally faster and cost-effective models, for reducing overfitting in the models and for providing insight and better understanding of underlying process of the dataset used for building the models.

Feature selection involves three main method types which are filter method, wrapper method and embedded method. In filter methods, features are selected based on the features' scores ranking in various statistical tests and the correlation results with outcome variable (example; Pearson's correlation, ANOVA). In wrapper methods, different subsets of features are generated, and each subset is used to build models and train learning algorithm. Features are added or removed from the subset based in the inferences from the trained model and the subset is selected for the test algorithm

(example; Forward and Backward selection). Embedded methods are combination of both filter and wrapper methods (example; lasso methods).

The following techniques are used for feature selection in this project; information gain, random forest and lasso regression.

#### 2.2.2.1 Information Gain:

Information Gain, which is also referred to as Mutual Information (MI) measures the dependency between two variables. It can be defined as the amount of information obtained about one random variable from observing the other random variable. As a simplified example, for independent variables X and Y, observing X will not provide information about Y and vice versa, this means that their mutual information is zero. On the other hand, if variables X and Y are dependent, observing the value of X will provide information about the value of Y.

For a pair of random variables (X, Y), if their joint distribution is $P(X,Y)$ and the marginal distributions are $P_X$ and $P_Y$, the mutual information is define as;

$$I(X;Y) = D_{KL}(P(X,Y) \;||\; P_X \otimes P_Y)$$

$$where \bigotimes = denotes\ the\ product\ measure, P_X \otimes P_Y$$

For the purpose of feature selection, information gain measures the dependence between dataset variables and the target variable. Feature selection is carried out on the dataset using selection of top ranking features having highest mutual information with target variable of the dataset, and figure 7 below represent selected 94 features which are significantly better than other features for prediction of target variables;

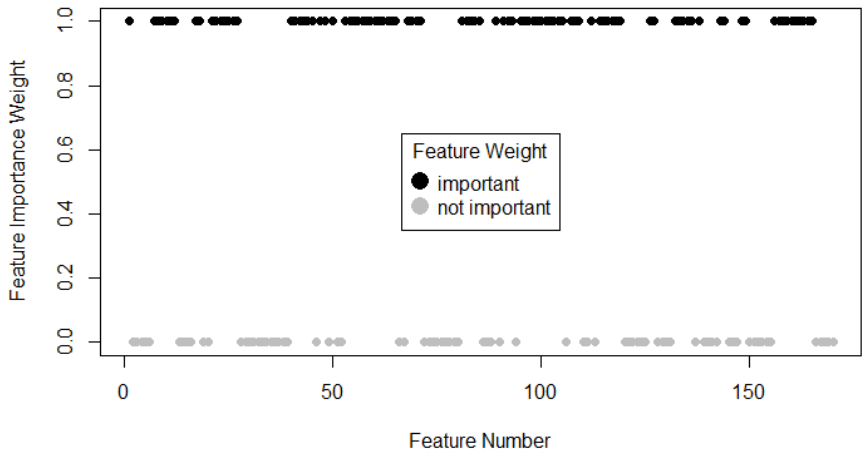


Figure 7: A plot showing weight of features using Information Gain

The figure 8 below represents the list of important features from dataset in accordance to the weight from information gain.

```
 [1] "am_0"    "al_000" "bj_000" "ag_002" "aq_000" "dn_000" "ap_000" "cn_000" "bh_000" "bb_000" "bu_000"
[12] "bv_000" "cq_000" "ck_000" "cj_000" "ah_000" "bg_000" "ee_005" "an_000" "ci_000" "ag_001" "ao_000"
[23] "ag_003" "aa_000" "bt_000" "cn_001" "cc_000" "bx_000" "ay_008" "az_001" "cs_002" "bi_000" "az_002"
[34] "cs_004" "ba_008" "az_005" "by_000" "ag_004" "ee_000" "dd_000" "az_000" "cv_000" "cn_004" "cm_000"
[45] "ba_009" "ba_000" "dc_000" "ce_000" "ds_000" "ag_005" "ee_001" "ee_002" "cn_002" "ba_004" "de_000"
[56] "cl_000" "dt_000" "az_007" "cn_003" "ba_002" "ba_003" "cs_003" "ay_009" "ba_001" "ba_005" "cs_005"
[67] "cx_000" "ee_004" "ba_006" "dg_000" "ee_006" "cs_001" "ed_000" "ec_00"  "ba_007" "ee_003" "bc_000"
[78] "cg_000" "di_000" "cn_008" "cs_000" "cn_007" "df_000" "ay_007" "cf_000" "cn_009" "bd_000" "ar_000"
[89] "eb_000" "ag_000" "do_000" "ai_000" "cn_005" "az_004"
```

Figure 8: List of important features using Information Gain

#### 2.2.2.2 Random Forest:

Random forest is one of the machine learning methods that are called ensemble learning methods and it consists of a number of decision trees. It can provide high prediction accuracy, low overfitting and easy interpretability, as compared to individual decision trees. They can increase the prediction accuracy through corrections of the instability of an individual decision tree by making small changes in learning sample.

Random forest works by drawing several bootstrap samples from the original dataset and each bootstrap sample is used to create an unpruned decision tree. The variable selected for each split in the decision tree is chosen from a small random subset of all the variables of the dataset, and this prevents the problem of “small n large p”.

Random forest produces decorrelated trees because randomness does not allow any tree in the forest to use all the variables or all the observations. Using random forest, the

value of the class variable is determined as the average or the majority vote of the predictions of all the decision trees.

Random forest can be used for a feature selection purpose and for a classification task, the measure of impurity for choosing the best features can either be Gini impurity or information gain/entropy while for a regression task, the measure of impurity is variance. When training a tree, the contribution of each feature in reducing the weighted impurity in the tree can be computed and the more a feature reduces the impurity, the more important the feature is. For random forests, the impurity reduction from each feature can be averaged to determine the final importance of the features.

Based on the feature importances computed through random forest using mean decreasing Gini, a threshold was applied to select features with mean decreasing Gini at least 5 and the figures 9 and 10 below indicate that 30 features are important out of the total 170 features in the sense that their mean decrease of Gini above the threshold:

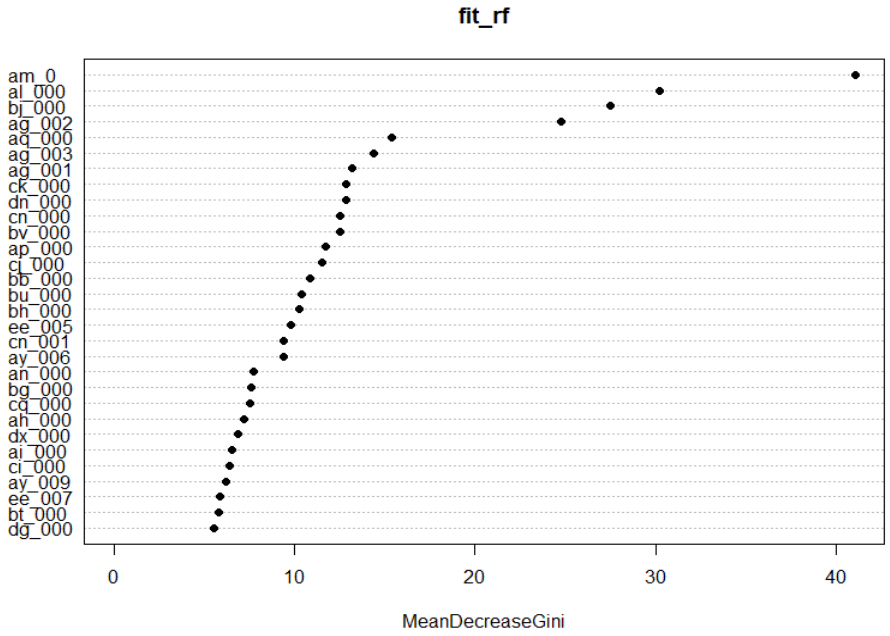


Figure 9: A plot important features vs level of importance using Random Forest

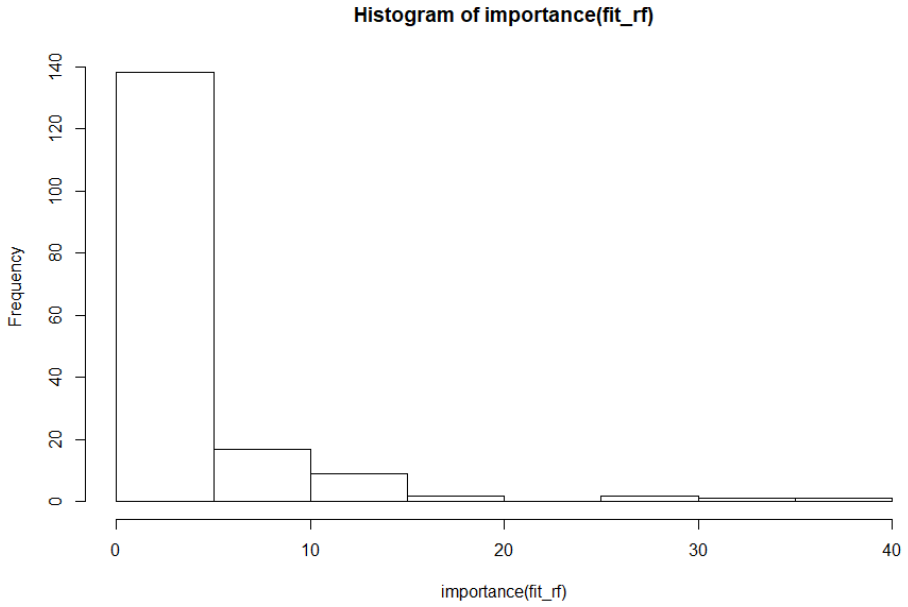


Figure 10: Histogram of feature importance (mean decrease of Gini impurity by including the feature) using Random Forest

#### 2.2.2.3 LASSO method:

LASSO (Least Absolute Shrinkage and Selection Operator) is a regularisation method which is used to reduce the model complexity and a powerful technique for feature selection by selecting the significant features to predict the dependent variables while shrinking the coefficients of unimportant features to zero. Lasso is L1 regularization method that puts a constraint on the sum of absolute values of the parameters of the model so that the sum needs to be less than a fixed value (upper bound). The total cost function with loss L is:

$$L + \lambda \sum |\beta_1|$$

The lambda($\lambda$) is a tuning parameter that controls the strength of the penalty and it is selected in a way that produces a model with minimal sample errors. The glmnet package in R implement the combined version of L1 and L2 regularization method called Elastic Net with the formula shown below; and the Lasso method can be carried out by setting the alpha($\alpha$) value to 1.

$$L + \lambda[(1-\alpha)\sum \beta_1^2 + \alpha \sum |\beta_1|]$$

The optimal value for lambda($\lambda$) is found by performing a grid search with cross-validation using cv.glmnet function in R and the result was shown in the figure below;

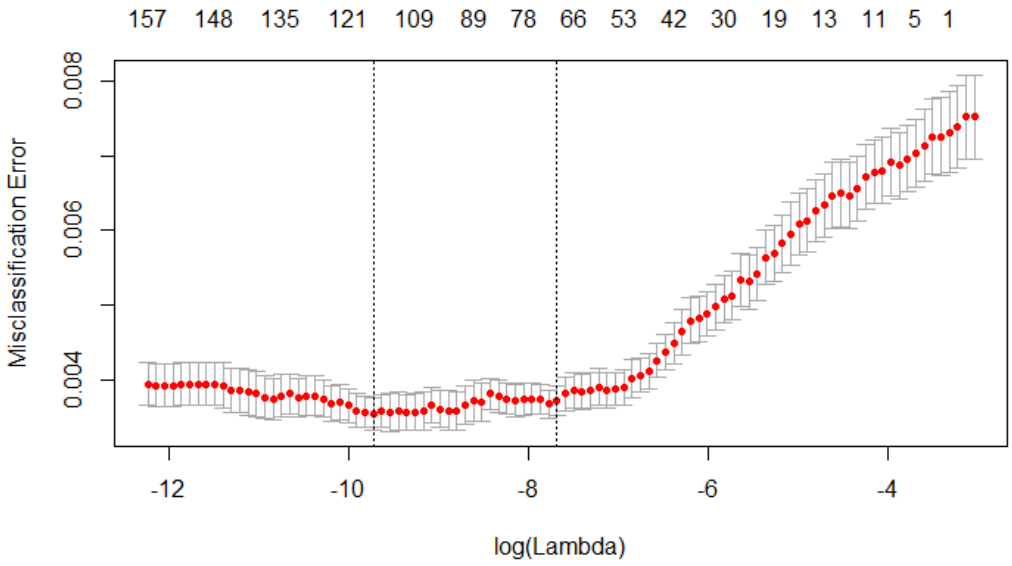


Figure 11: A plot to find optimal value of lambda($\lambda$)

The first vertical line is the log of optimal value of lambda($\lambda$) that minimises the misclassification error and the exact value can be found by using lambda_min. While the second vertical line is the log of optimal value of lambda($\lambda$) that balances accuracy with model simplicity that is producing model with only the important features and the exact value can be found using lambda_1se. The Lasso features selection method produced 68 features which balance accuracy with model simplicity out of total 170 features in the dataset.

# Chapter Three

This project uses a classification method which involves techniques for determining the class of target variable using one or more features. This section describes the measures used to determine the classification model performance and two classification models are examined using the dataset.

## 3.1 Model Evaluation Metrics

It is important to evaluate the machine learning classification model to determine the prediction performance of the model to the new unseen observations. There are various model evaluation metrics that can be used for checking prediction performance of a classification model, and in this project the following commonly used model evaluation metrics are considered; accuracy, precision, recall, F1 score, and area under curve of receiver operating characteristics (AUCROC). The classification models for this project are binary classification where the target variable has only two classes to be predicted and straightforward explanation of evaluation metrics such as accuracy, precision, recall and F1 score can be achieved using confusion matrix.

Confusion matrix

In case of binary classification, confusion matrix can be defined as table with four different combinations of actual values and predicted values. It is a table that provides information on the performance of classification model on the prediction of values of test dataset against the true values. Confusion matrix provides information about errors made by the classification model and most importantly, the values of the types of errors, that is, type I and type II errors. Figure 12 represents confusion matrix showing information about recall, precision, accuracy, and F1 score;

| | | Predicted Class | | |
|---|---|---|---|---|
| | | Positive | Negative | |
| Actual Class | Positive | **TP** | **FN** (Type II error) | **Recall / Sensitivity** |
| | Negative | **FP** (Type I error) | **TN** | **Specificity** |
| **Accuracy** | | **Precision** | | **F1 score** |

Figure 12: confusion matrix

Definition of terms:

**TP**: This means that the class of data object is positive, and the predicted class positive.

**TN**: This means that the class of data object is negative, and the predicted class negative.

**FP**: This means that the class of data object is negative, and the predicted class is positive. This refers to type I error.

**FN**: This means that the class of data object is positive, and the predicted class is negative. This refers to type II error.

**Sensitivity**: This refers to as recall, and it represents the ratio of correctly predicted positive data objects to the total actual positive data objects.

$$Sensitivity, = \frac{TP}{TP + FN}$$

**Specificity**: This represents the ratio of correctly predicted negative data objects to the total actual negative data objects.

$$Specificity, = \frac{TN}{FP + TN}$$

Definition of metrics:

**Accuracy**: This represents the ratio of correctly predicted data objects to the total data objects.

$$Accuracy, = \frac{TP + TN}{TP + TN + FP + FN}$$

**Precision**: This represents the ratio of correctly predicted positive data objects to the total predicted positive data objects. Precision represents measure of exactness or quality.

$$Precision, = \frac{TP}{TP + FP}$$

**Recall**: This represents the ratio of correctly predicted positive data objects to the total actual positive data objects. Recall represents measure of completeness or quantity.

$$Recall, = \frac{TP}{TP + FN}$$

**F1 score**: This represents the harmonic mean of both precision and recall measurements.

$$F1\ score, = \frac{2 * Recall * Precision}{Recall + Precision}$$

**AUCROC**: The ROC curve is defined as the probability curve where the true positive rate (sensitivity) is plotted against false positive rate(100-specificity) at various cut off points, while AUC can be defined as the measure of separability of class, and the higher the value of AUC, the better the model in predicting positive as positive and negative as negative.

This is used to measure model performance using all possible probability cutoffs. The starting point of AUC is 0.5 which represents that the model is doing guess work in predicting class label, and the closer the AUC is to 1, the better the classifier is in predicting correct class label. The Figure below represent the concept of AUCROC.

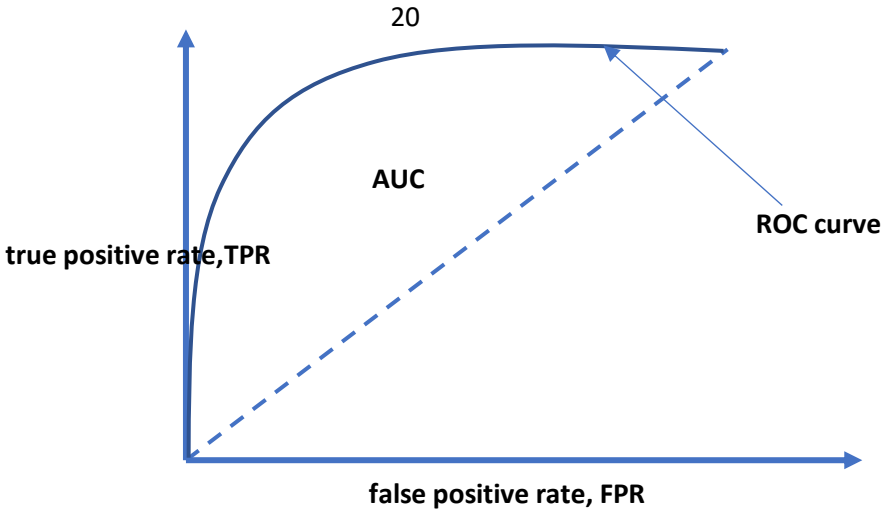


Figure 13: Concept of ROC curve

## 3.2 Classification Models

Selection of an appropriate classification model for a task can be challenging because each classification algorithm has its own characteristics and certain assumptions, which make it difficult for one classification model to work best across all possible situations. In the remaining part of this section, we are comparing the performance of two classification models which are logistics regression and Naïve Bayes classifiers with different set of input variables.

### 3.2.1 Logistic regression

Logistics regression is a statistical model that is used to predict the probability of occurrence of a certain class such as pass/fail, default/not default, and win/lose. It is an extension version of linear regression which is used to solve classification problems where the dependent variable is categorical variable. Depending on the numbers and order of classes of the dependent variable, logistic regression can be classified as binary logistic regression where there are two number of classes in the dependent variable, multinomial logistics regression where there are more than two numbers of classes in the dependent variable, and ordinal logistic regression where there are more than two number of ordered classes in the dependent variable.

In logistic regression models, the logarithm of odds (also refers to as log-odds) for the class labelled “1” is the linear combination of all the independent variables. Odds is the defined as the probability of occurrence of an event divided by probability of no

occurrence of the event. It is assumed that the threshold for classification task is 0.5 while the outcome of a class label is a probability between 0 and 1 inclusively.

The logistic function is mathematically defined as;

$$f(\eta) = \frac{1}{1+ e^{-\eta}} \quad (1)$$

The logarithm of odds is mathematically defined as;

$$\ln\left(\frac{P(y=1)}{1-P(y=1)}\right) = \beta_0 + \beta_1 x_1 + --- + \beta_p x_p \quad (2)$$

The relationship and transformation from linear regression to logistic regression resulted into the formula below where the linear combination is wrapped inside the right section of the logistic function and it can also be formulated by taking the exponential function of log-odds.

$$P(y=1) = \frac{1}{1+e^{-(\beta_0+ \beta_1 x_1 + --- + \beta_p x_p)}} \quad (3)$$

Binary logistic regression model is used in this project because the target label is categorical variable with labelled 1 and 0. Four different types of binary logistic regression models were created which are logistic regression with all features in the data, logistic regressions with features selected from information gain, random forest and lasso regression feature selection techniques.

The models were built, and the total sample sizes were randomly divided into training set and test set through 'caTools' package in R. The total sample size was 44667, out of which 70% were selected as training set and remaining 30% as test set. The performances of the models were evaluated and compared. Table 3.1 below show the performances of the different models with their evaluation metrics;

| | All features (171) | I.G features (94) | R.F features (30) | Lasso Reg. features (68) |
|---|---|---|---|---|
| Accuracy | 0.9947 | 0.9952 | 0.9951 | 0.9962 |
| Precision | 0.6667 | 0.7176 | 0.7500 | 0.8400 |
| Recall | 0.5941 | 0.5980 | 0.5347 | 0.6238 |
| F1 score | 0.6283 | 0.6524 | 0.6243 | 0.7159 |
| AUCROC | 0.7959 | 0.8010 | 0.9659 | 0.9443 |

Table 3.1: Logistics regression models

| | Estimate | Std. Error | z value | Pr(>\|z\|) |
|---|---|---|---|---|
| (Intercept) | -1.008e+15 | 1.296e+06 | -7.780e+08 | <2e-16 *** |
| aa_000 | -7.872e+13 | 6.598e+05 | -1.193e+08 | <2e-16 *** |
| ab_000 | 7.613e+12 | 1.375e+05 | 5.535e+07 | <2e-16 *** |
| ac_000 | -1.785e+05 | 5.185e-04 | -3.442e+08 | <2e-16 *** |
| ad_000 | 2.895e+09 | 6.139e+02 | 4.716e+06 | <2e-16 *** |
| ae_000 | -6.060e+10 | 3.937e+03 | -1.539e+07 | <2e-16 *** |
| af_000 | 5.051e+10 | 3.216e+03 | 1.571e+07 | <2e-16 *** |
| ag_000 | 6.579e+08 | 1.531e+01 | 4.296e+07 | <2e-16 *** |
| ag_001 | -3.910e+09 | 7.313e+01 | -5.346e+07 | <2e-16 *** |
| ag_002 | 2.647e+07 | 2.224e+01 | 1.190e+06 | <2e-16 *** |
| ag_003 | 1.567e+08 | 4.729e+00 | 3.314e+07 | <2e-16 *** |
| ag_004 | -3.253e+07 | 2.001e+00 | -1.626e+07 | <2e-16 *** |
| ag_005 | -1.845e+08 | 1.593e+00 | -1.159e+08 | <2e-16 *** |
| ag_006 | 1.584e+08 | 1.534e+00 | 1.032e+08 | <2e-16 *** |
| ag_007 | -9.981e+08 | 2.218e+00 | -4.501e+08 | <2e-16 *** |
| ag_008 | 1.158e+09 | 7.032e+00 | 1.646e+08 | <2e-16 *** |
| ag_009 | -5.449e+08 | 3.950e+00 | -1.379e+08 | <2e-16 *** |
| ah_000 | -1.687e+09 | 1.156e+02 | -1.459e+07 | <2e-16 *** |
| ai_000 | -3.643e+08 | 7.644e+00 | -4.765e+07 | <2e-16 *** |
| aj_000 | 5.215e+09 | 7.598e+01 | 6.863e+07 | <2e-16 *** |
| ak_000 | -2.278e+09 | 3.100e+01 | -7.348e+07 | <2e-16 *** |

| | | | | | |
|---|---|---|---|---|---|
| al_000 | -1.685e+09 | 1.682e+01 | -1.002e+08 | <2e-16 | *** |
| am_0 | 1.010e+09 | 1.027e+01 | 9.837e+07 | <2e-16 | *** |
| an_000 | -1.085e+08 | 2.487e+00 | -4.364e+07 | <2e-16 | *** |
| ao_000 | -1.359e+07 | 2.075e+00 | -6.549e+06 | <2e-16 | *** |
| ap_000 | 1.487e+09 | 4.391e+01 | 3.386e+07 | <2e-16 | *** |
| aq_000 | 3.464e+08 | 4.669e+00 | 7.419e+07 | <2e-16 | *** |
| ar_000 | -1.157e+12 | 8.387e+04 | -1.380e+07 | <2e-16 | *** |
| as_000 | -2.568e+09 | 4.656e+01 | -5.515e+07 | <2e-16 | *** |
| at_000 | -8.904e+07 | 5.989e+00 | -1.487e+07 | <2e-16 | *** |
| au_000 | 6.072e+09 | 5.069e+01 | 1.198e+08 | <2e-16 | *** |
| av_000 | 1.899e+10 | 1.517e+02 | 1.252e+08 | <2e-16 | *** |
| ax_000 | -1.050e+11 | 6.393e+02 | -1.642e+08 | <2e-16 | *** |
| ay_000 | -7.386e+08 | 1.637e+01 | -4.513e+07 | <2e-16 | *** |
| ay_001 | 6.528e+08 | 3.457e+01 | 1.889e+07 | <2e-16 | *** |
| ay_002 | 1.684e+09 | 5.820e+01 | 2.893e+07 | <2e-16 | *** |
| ay_003 | -1.733e+08 | 1.550e+01 | -1.118e+07 | <2e-16 | *** |
| ay_004 | -8.751e+08 | 2.213e+01 | -3.955e+07 | <2e-16 | *** |
| ay_005 | 9.414e+07 | 9.022e-01 | 1.043e+08 | <2e-16 | *** |
| ay_006 | -9.038e+07 | 1.392e+00 | -6.492e+07 | <2e-16 | *** |
| ay_007 | -3.922e+07 | 1.326e+00 | -2.958e+07 | <2e-16 | *** |
| ay_008 | -1.682e+07 | 1.298e+00 | -1.296e+07 | <2e-16 | *** |
| ay_009 | 2.495e+08 | 7.477e+00 | 3.337e+07 | <2e-16 | *** |
| az_000 | 3.104e+08 | 1.357e+01 | 2.287e+07 | <2e-16 | *** |
| az_001 | -4.997e+09 | 7.787e+01 | -6.417e+07 | <2e-16 | *** |
| az_002 | 3.691e+08 | 8.018e+01 | 4.604e+06 | <2e-16 | *** |
| az_003 | -1.489e+07 | 2.370e+00 | -6.285e+06 | <2e-16 | *** |
| az_004 | -1.419e+07 | 1.272e+00 | -1.116e+07 | <2e-16 | *** |
| az_005 | -5.558e+06 | 1.258e+00 | -4.418e+06 | <2e-16 | *** |
| az_006 | -1.379e+08 | 1.473e+00 | -9.357e+07 | <2e-16 | *** |
| az_007 | 2.249e+08 | 4.662e+00 | 4.825e+07 | <2e-16 | *** |
| az_008 | -7.149e+08 | 4.073e+01 | -1.755e+07 | <2e-16 | *** |
| az_009 | 3.623e+09 | 1.909e+02 | 1.898e+07 | <2e-16 | *** |
| ba_000 | -3.353e+07 | 2.521e+00 | -1.330e+07 | <2e-16 | *** |

| | | | | | |
|---|---|---|---|---|---|
| ba_001 | -1.841e+08 | 2.528e+00 | -7.282e+07 | <2e-16 | *** |
| ba_002 | -2.002e+08 | 4.792e+00 | -4.177e+07 | <2e-16 | *** |
| ba_003 | -8.336e+08 | 1.080e+01 | -7.721e+07 | <2e-16 | *** |
| ba_004 | 4.767e+07 | 1.243e+01 | 3.835e+06 | <2e-16 | *** |
| ba_005 | 8.379e+07 | 7.775e+00 | 1.078e+07 | <2e-16 | *** |
| ba_006 | -9.552e+08 | 4.825e+00 | -1.980e+08 | <2e-16 | *** |
| ba_007 | 4.477e+08 | 3.824e+00 | 1.171e+08 | <2e-16 | *** |
| ba_008 | 6.005e+08 | 8.875e+00 | 6.766e+07 | <2e-16 | *** |
| ba_009 | -1.100e+07 | 5.060e+00 | -2.173e+06 | <2e-16 | *** |
| bb_000 | -1.539e+12 | 1.242e+05 | -1.239e+07 | <2e-16 | *** |
| bc_000 | 1.503e+10 | 1.673e+02 | 8.982e+07 | <2e-16 | *** |
| bd_000 | 1.213e+09 | 1.511e+02 | 8.025e+06 | <2e-16 | *** |
| be_000 | -9.142e+09 | 7.120e+01 | -1.284e+08 | <2e-16 | *** |
| bf_000 | 9.119e+10 | 1.108e+03 | 8.230e+07 | <2e-16 | *** |
| bg_000 | 1.671e+09 | 1.156e+02 | 1.445e+07 | <2e-16 | *** |
| bh_000 | -8.835e+09 | 4.664e+01 | -1.894e+08 | <2e-16 | *** |
| bi_000 | -1.728e+09 | 4.291e+01 | -4.028e+07 | <2e-16 | *** |
| bj_000 | -1.837e+09 | 4.287e+01 | -4.285e+07 | <2e-16 | *** |
| bk_000 | -5.600e+07 | 1.663e+00 | -3.368e+07 | <2e-16 | *** |
| bl_000 | -1.362e+09 | 1.163e+00 | -1.171e+09 | <2e-16 | *** |
| bm_000 | 3.342e+08 | 9.358e-01 | 3.572e+08 | <2e-16 | *** |
| bn_000 | 1.119e+09 | 2.227e+00 | 5.023e+08 | <2e-16 | *** |
| bo_000 | -2.235e+08 | 2.403e+00 | -9.300e+07 | <2e-16 | *** |
| bp_000 | -9.407e+08 | 3.734e+00 | -2.519e+08 | <2e-16 | *** |
| bq_000 | 9.601e+08 | 4.772e+00 | 2.012e+08 | <2e-16 | *** |
| br_000 | -8.957e+08 | 3.159e+00 | -2.835e+08 | <2e-16 | *** |
| bs_000 | 1.830e+09 | 1.022e+01 | 1.791e+08 | <2e-16 | *** |
| bt_000 | 7.872e+13 | 6.598e+05 | 1.193e+08 | <2e-16 | *** |
| bu_000 | 7.303e+11 | 1.294e+04 | 5.645e+07 | <2e-16 | *** |
| bv_000 | -1.792e+12 | 1.478e+05 | -1.212e+07 | <2e-16 | *** |
| bx_000 | -4.487e+07 | 6.226e-01 | -7.207e+07 | <2e-16 | *** |
| by_000 | 2.066e+09 | 9.402e+01 | 2.197e+07 | <2e-16 | *** |
| bz_000 | -3.829e+08 | 5.421e+00 | -7.063e+07 | <2e-16 | *** |
| ca_000 | -4.906e+07 | 2.146e+01 | -2.286e+06 | <2e-16 | *** |

| | | | | | |
|---|---|---|---|---|---|
| cb_000 | -2.076e+08 | 2.906e+00 | -7.144e+07 | <2e-16 | *** |
| cc_000 | 5.018e+07 | 7.159e-01 | 7.009e+07 | <2e-16 | *** |
| cd_000 | NA | NA | NA | NA | |
| ce_000 | -1.739e+09 | 1.348e+01 | -1.290e+08 | <2e-16 | *** |
| cf_000 | 2.312e+09 | 5.825e+02 | 3.969e+06 | <2e-16 | *** |
| cg_000 | 7.055e+10 | 1.718e+03 | 4.107e+07 | <2e-16 | *** |
| ch_000 | -1.394e+14 | 1.277e+07 | -1.092e+07 | <2e-16 | *** |
| ci_000 | 3.598e+07 | 6.881e-01 | 5.228e+07 | <2e-16 | *** |
| cj_000 | 8.943e+07 | 1.147e+00 | 7.797e+07 | <2e-16 | *** |
| ck_000 | -2.742e+07 | 1.179e+00 | -2.326e+07 | <2e-16 | *** |
| cl_000 | -2.582e+10 | 1.473e+02 | -1.753e+08 | <2e-16 | *** |
| cm_000 | 8.533e+10 | 6.865e+02 | 1.243e+08 | <2e-16 | *** |
| cn_000 | 4.029e+08 | 2.512e+01 | 1.604e+07 | <2e-16 | *** |
| cn_001 | 1.109e+08 | 1.152e+01 | 9.624e+06 | <2e-16 | *** |
| cn_002 | 1.493e+08 | 3.600e+00 | 4.147e+07 | <2e-16 | *** |
| cn_003 | 1.282e+08 | 2.621e+00 | 4.893e+07 | <2e-16 | *** |
| cn_004 | 1.286e+08 | 2.063e+00 | 6.234e+07 | <2e-16 | *** |
| cn_005 | 8.166e+07 | 2.439e+00 | 3.348e+07 | <2e-16 | *** |
| cn_006 | -4.486e+06 | 2.752e+00 | -1.630e+06 | <2e-16 | *** |
| cn_007 | 3.429e+08 | 6.639e+00 | 5.165e+07 | <2e-16 | *** |
| cn_008 | 1.039e+08 | 9.918e+00 | 1.048e+07 | <2e-16 | *** |
| cn_009 | -2.533e+08 | 1.188e+01 | -2.132e+07 | <2e-16 | *** |
| co_000 | -5.207e+09 | 2.594e+02 | -2.007e+07 | <2e-16 | *** |
| cp_000 | 8.123e+09 | 1.531e+02 | 5.306e+07 | <2e-16 | *** |
| cq_000 | 2.601e+12 | 1.940e+05 | 1.340e+07 | <2e-16 | *** |
| cr_000 | 2.048e+10 | 2.758e+02 | 7.428e+07 | <2e-16 | *** |
| cs_000 | -1.705e+10 | 1.115e+02 | -1.529e+08 | <2e-16 | *** |
| cs_001 | 5.100e+10 | 1.806e+03 | 2.824e+07 | <2e-16 | *** |
| cs_002 | 5.064e+07 | 3.521e+00 | 1.438e+07 | <2e-16 | *** |
| cs_003 | 2.541e+08 | 3.605e+00 | 7.050e+07 | <2e-16 | *** |
| cs_004 | 2.008e+08 | 2.851e+00 | 7.042e+07 | <2e-16 | *** |
| cs_005 | 1.726e+08 | 2.781e+00 | 6.207e+07 | <2e-16 | *** |
| cs_006 | 1.533e+08 | 2.865e+00 | 5.349e+07 | <2e-16 | *** |
| cs_007 | 4.710e+08 | 8.663e+00 | 5.437e+07 | <2e-16 | *** |

| | | | | | |
|---|---|---|---|---|---|
| cs_008 | -5.554e+09 | 7.151e+01 | -7.766e+07 | <2e-16 | *** |
| cs_009 | -2.196e+10 | 5.569e+02 | -3.943e+07 | <2e-16 | *** |
| ct_000 | 7.167e+09 | 2.557e+02 | 2.803e+07 | <2e-16 | *** |
| cu_000 | 1.844e+10 | 2.452e+02 | 7.520e+07 | <2e-16 | *** |
| cv_000 | 1.692e+08 | 7.060e-01 | 2.397e+08 | <2e-16 | *** |
| cx_000 | -1.018e+08 | 8.582e-01 | -1.186e+08 | <2e-16 | *** |
| cy_000 | -3.877e+09 | 5.249e+01 | -7.386e+07 | <2e-16 | *** |
| cz_000 | 1.197e+09 | 6.574e+00 | 1.821e+08 | <2e-16 | *** |
| da_000 | -4.179e+11 | 2.650e+03 | -1.577e+08 | <2e-16 | *** |
| db_000 | 1.145e+10 | 8.731e+03 | 1.311e+06 | <2e-16 | *** |
| dc_000 | -1.471e+08 | 7.352e-01 | -2.000e+08 | <2e-16 | *** |
| dd_000 | 1.726e+10 | 1.283e+02 | 1.345e+08 | <2e-16 | *** |
| de_000 | -2.344e+09 | 3.376e+02 | -6.942e+06 | <2e-16 | *** |
| df_000 | 1.842e+07 | 7.082e+00 | 2.601e+06 | <2e-16 | *** |
| dg_000 | 2.760e+08 | 4.250e+00 | 6.494e+07 | <2e-16 | *** |
| dh_000 | 2.070e+08 | 1.734e+01 | 1.193e+07 | <2e-16 | *** |
| di_000 | -3.165e+07 | 1.520e+00 | -2.082e+07 | <2e-16 | *** |
| dj_000 | -6.081e+11 | 4.133e+03 | -1.471e+08 | <2e-16 | *** |
| dk_000 | 4.221e+08 | 6.508e+00 | 6.486e+07 | <2e-16 | *** |
| dl_000 | 8.253e+07 | 1.156e+00 | 7.138e+07 | <2e-16 | *** |
| dm_000 | -4.021e+08 | 5.431e+00 | -7.404e+07 | <2e-16 | *** |
| dn_000 | 1.219e+10 | 8.959e+01 | 1.360e+08 | <2e-16 | *** |
| do_000 | 5.796e+09 | 3.137e+01 | 1.848e+08 | <2e-16 | *** |
| dp_000 | -4.334e+10 | 1.142e+02 | -3.796e+08 | <2e-16 | *** |
| dq_000 | -1.890e+05 | 9.133e-03 | -2.069e+07 | <2e-16 | *** |
| dr_000 | -5.853e+07 | 1.067e+00 | -5.487e+07 | <2e-16 | *** |
| ds_000 | 3.419e+08 | 1.180e+01 | 2.897e+07 | <2e-16 | *** |
| dt_000 | 5.925e+09 | 6.872e+01 | 8.621e+07 | <2e-16 | *** |
| du_000 | -2.019e+07 | 2.045e-01 | -9.872e+07 | <2e-16 | *** |
| dv_000 | 1.320e+08 | 1.828e+00 | 7.217e+07 | <2e-16 | *** |
| dx_000 | 1.273e+06 | 2.479e-01 | 5.134e+06 | <2e-16 | *** |
| dy_000 | -1.068e+09 | 1.432e+01 | -7.457e+07 | <2e-16 | *** |
| dz_000 | 2.591e+12 | 4.643e+04 | 5.580e+07 | <2e-16 | *** |
| ea_000 | -6.162e+11 | 7.860e+03 | -7.839e+07 | <2e-16 | *** |

| | | | | |
|---|---|---|---|---|
| eb_000 | 6.796e+05 | 1.532e-02 | 4.437e+07 | <2e-16 *** |
| ec_00 | -7.985e+10 | 6.654e+02 | -1.200e+08 | <2e-16 *** |
| ed_000 | 5.400e+10 | 9.935e+02 | 5.435e+07 | <2e-16 *** |
| ee_000 | -7.731e+07 | 3.384e+00 | -2.285e+07 | <2e-16 *** |
| ee_001 | -2.526e+08 | 3.332e+00 | -7.582e+07 | <2e-16 *** |
| ee_002 | 5.420e+08 | 4.638e+00 | 1.169e+08 | <2e-16 *** |
| ee_003 | -1.156e+09 | 6.844e+00 | -1.689e+08 | <2e-16 *** |
| ee_004 | -1.104e+07 | 3.611e+00 | -3.058e+06 | <2e-16 *** |
| ee_005 | 3.952e+08 | 3.274e+00 | 1.207e+08 | <2e-16 *** |
| ee_006 | -4.446e+07 | 3.092e+00 | -1.438e+07 | <2e-16 *** |
| ee_007 | -1.787e+07 | 2.994e+00 | -5.968e+06 | <2e-16 *** |
| ee_008 | -3.188e+08 | 4.265e+00 | -7.474e+07 | <2e-16 *** |
| ee_009 | -2.127e+09 | 1.790e+01 | -1.188e+08 | <2e-16 *** |
| ef_000 | -3.465e+13 | 9.235e+04 | -3.752e+08 | <2e-16 *** |
| eg_000 | 1.511e+12 | 4.003e+04 | 3.776e+07 | <2e-16 *** |

Table 3.2: Coefficients of all features with Logistic Regression

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 2766.9 on 31266 degrees of freedom
Residual deviance: 8290.0 on 31097 degrees of freedom
AIC: 8630

The result of logistic regression model using all the features in the dataset described that all the features, except feature cd_000 which is constant feature, were statistically significant as shown in Table 3.2. The evaluation metrics were compared amongst models with different selected features and the selection techniques with highest evaluation metrics were highlighted with green colors. The table shows that models constructed with features from Lasso regression feature selection techniques had the highest values in terms of accuracy, precision, recall and F1score, while model constructed with features from random forest feature selection technique had the highest value in terms of AUCROC. Thus, logistic regression model demonstrated a high level of performance with models built through Lasso regression features. The Figures below represent the ROC curves of logistic regression models.

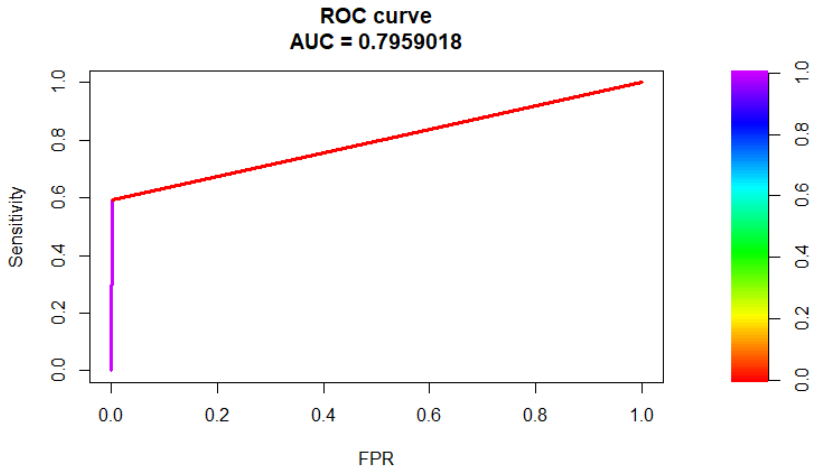


Figure 14: ROC curve – logistic regression (All features)

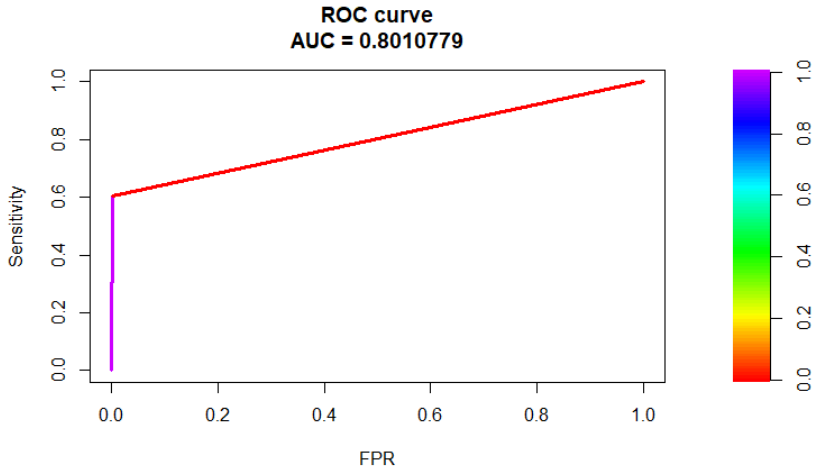


Figure 15: ROC curve – logistic regression with Information Gain (I.G) features

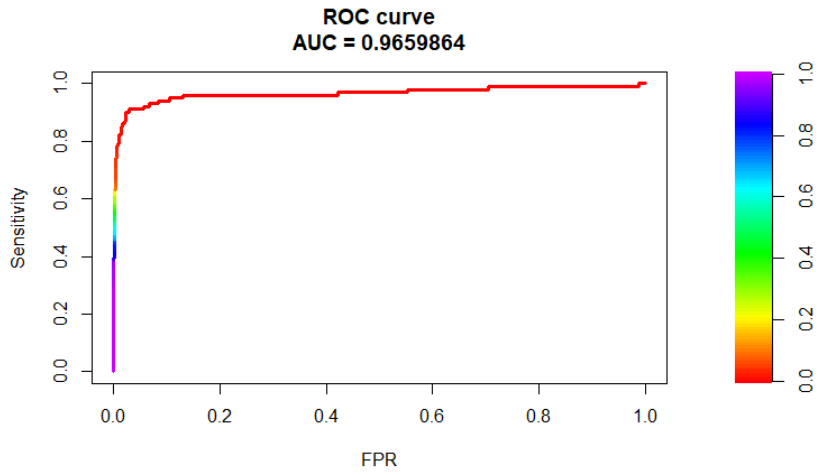


Figure 16: ROC curve – logistic regression with Random Forest (R.F) features

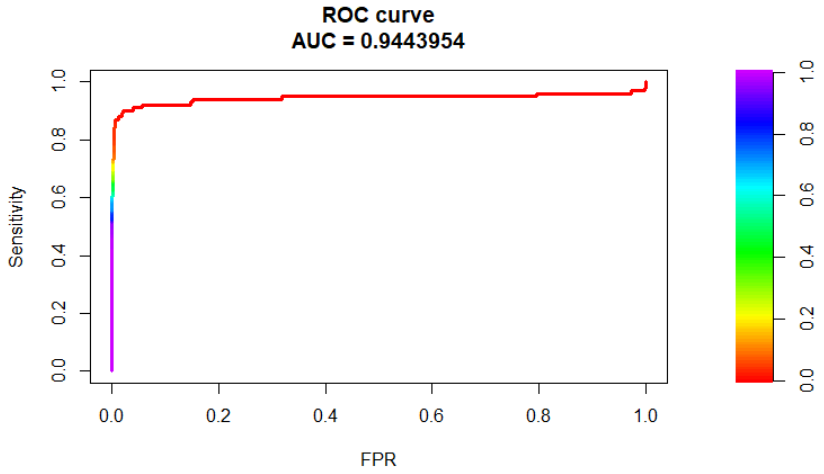


Figure 17: ROC curve – logistic regression with Lasso Regression (L.R) features

### 3.2.2 Naïve Bayes Classifier

Naïve Bayes classifier is one of the practical Bayesians learning methods where calculation for hypothesis is explicitly based on probabilities through application of Bayes theorem with the fundamental assumption that each feature makes an independent contribution to the result of the outcome. Bayes theorem provides a way of calculating the probability of occurrence of an event based on the probability of another event that has already occurred.

Bayes theorem is mathematically stated;

$$P(A|B) = \frac{P(B|A) * P(A)}{P(B)} \quad (4)$$

Where A and B represent event;

- P(B|A) represents the likelihood of event B given event A.
- P(B) is called evidence which represents the given training data.
- P(A) is called prior probability which represents probability of event before the evidence is seen.
- P(A|B) is called posterior probability which represent probability of event A after the evidence is seen.

According to Bayes theorem, P(A|B) increases as the values of P(B|A) and P(A) increases while P(A|B) decreases as the value of P(B) increases. In accordance to the classification task of this project, Bayes theorem can be applied to the dataset in the following form;

$$P(y|X) = \frac{P(X|y) * P(y)}{P(X)} \quad (5)$$

In equation (5), y represents the target variable and X represents vector of features of size k. With $X = x_1, x_2, \ldots\ldots, x_k$ substituted in the above equation, it resulted to the equation below;

$$P(y|x_1, x_2, .., x_k) = \frac{P(x_1|y)P(x_2|y)\ldots P(x_k|y)P(y)}{P(x_1)P(x_2)\ldots P(x_k)} \quad (6)$$

$$P(y|x_1, x_2, .., x_k) = \frac{P(y)\Pi_{i=1}^{k} P(x_i|y)}{P(x_1)P(x_2)\ldots P(x_k)} \quad (7)$$

The denominator term can be removed because it is a constant for any given input, and the interest is to find the maximum probability which can be expressed as;

$$y = argmax\ P(y)\Pi_{i=1}^{k} P(x_i|y) \quad (8)$$

Where $P(y)$ is called class probability and $P(x_i|y)$ is called conditional probability.

Four different types of binary Naïve Bayes classifiers were created which are a classifier with all features in the data, classifiers with features selected from information gain, random forest and lasso regression feature selection techniques. The classifiers were built through 'e1071' package in R, while total sample size was 44667, out of which 70% were selected as training set and remaining 30% as test set. The performances of the classifiers are evaluated and compared. The table below shows the performances of the different models with their evaluation metrics;

| | All features (171) | I.G features (94) | R.F features (30) | Lasso Reg. features (68) |
|---|---|---|---|---|
| Accuracy | 0.9694 | 0.9735 | 0.9811 | 0.9775 |
| Precision | 0.1828 | 0.2092 | 0.2683 | 0.2340 |
| Recall | 0.8812 | 0.9010 | 0.8713 | 0.8713 |
| F1 score | 0.3028 | 0.3396 | 0.4103 | 0.3689 |
| AUCROC | 0.9256 | 0.9375 | 0.9266 | 0.9248 |

Table 3.3: Naïve Bayes models

The evaluation metrics were compared amongst models with different selected features and the selection techniques with highest evaluation metrics were highlighted with green colors.

Table 3.3 shows that the model constructed with features from random forest feature selection techniques had the highest values in terms of accuracy, precision, and F1-score, while the model constructed with features from information gain feature selection technique had the highest value in terms of recall and AUCROC. Thus, Naïve Bayes model demonstrated a high level of performance with models built through random forest features.

However, it is important to highlight that generally the precisions and F1 scores from Naïve Bayes models were low compare to the ones from logistics regression classifiers

as shown in Tables 3.1, because of a relatively great false positive number. Thus, although recalls from Naïve Bayes models were much better than that from logistics regression classifiers as shown in Table 3.1, precisions were clearly worse than in Table 3.1. Figures 18-21 represents the ROC curves for each of Naïve Bayes models.

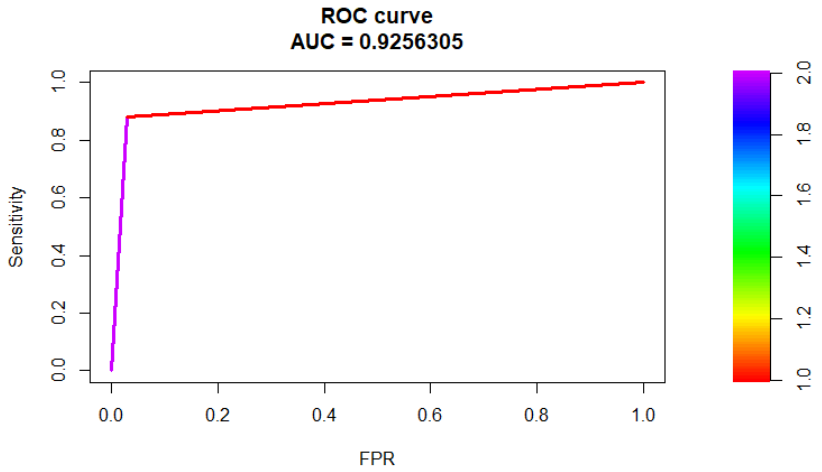


Figure 18: ROC curve – Naïve Bayes (All features)

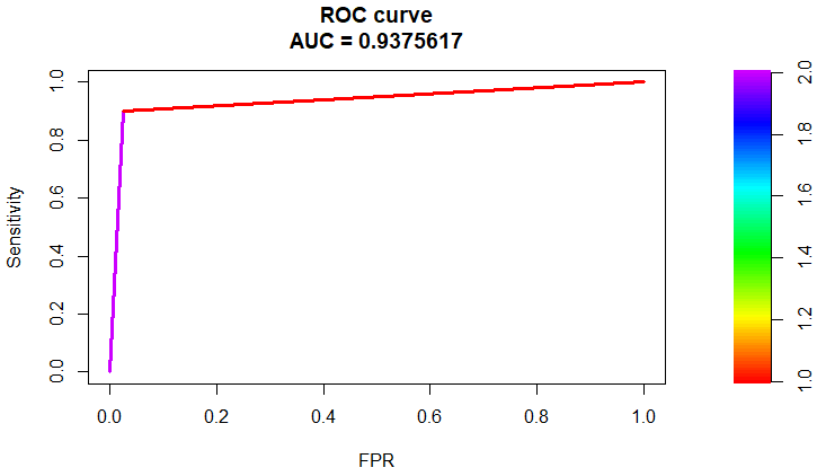


Figure 19: ROC curve – Naïve Bayes (I.G features)

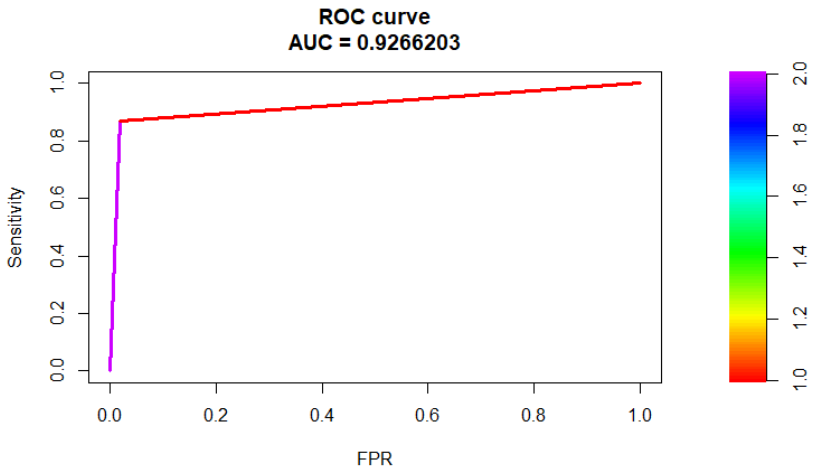


Figure 20: ROC curve – Naïve Bayes (R.F features)

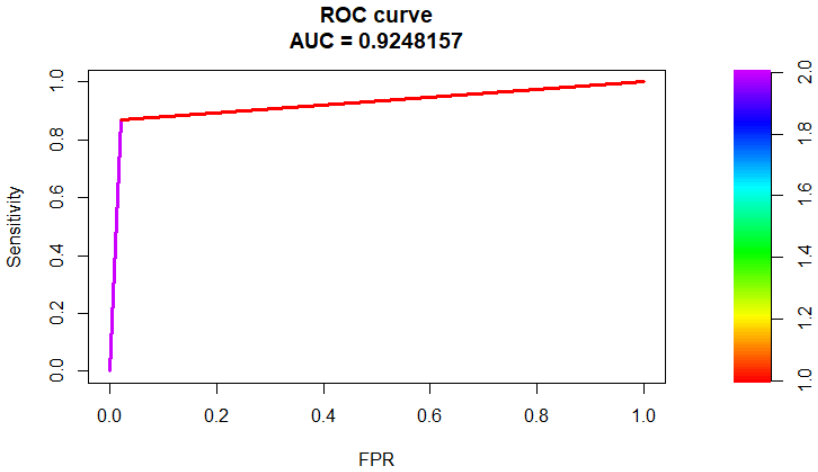


Figure 21: ROC curve – Naïve Bayes (L.R features)

# Chapter Four

This section involves using the dataset of this project to construct and examine the performances of more machine learning classification methods, such as K-nearest neighbor, support vector machines and ensemble learnings.

## 4.1 K - Nearest Neighbors classifier

K - Nearest Neighbors (also always refers to as KNN) is one of the instance-based learning methods where training dataset is stored and learning of discriminative function is delayed and carried out until there is a new instance to be classified. When there is a new instance, a set of instances that are like the new instance are retrieved from the stored training dataset and they are used to classify the new instance. This characteristic is one of the reasons KNN refers to as lazy learning method, and KNN belongs to the subcategory of non-parametric models which means the model does not make any underlying assumptions about the distribution of the dataset [8].

KNN model is simple to implement, robust to noise in the dataset and effective with large dataset because the classifier adapts as the new training dataset arrives. However, the computational cost for classifying a new instance increases as the number of samples in the training dataset increases because nearly all computation is executed at the classification time, unless the dataset has a very small dimensions (features) and the techniques for efficient indexing of training dataset are implemented. There is also a challenge of high storage cost with the large dataset [9].

It is important to indicate that KNN is vulnerable to overfitting due to the curse of dimensionality and this means that if the class of new instance depends on only a few features out of high dimensional features, then the closest neighbors may be at large distance apart and may not give accurate prediction. The concept of regularization can be used to avoid overfitting. However, in models where regularization cannot be applicable such as KNN, feature selection and dimensionality reduction techniques can be used to avoid the overfitting due to curse of dimensionality [9].

The similar objects to a new instance are selected through distance metric such as Euclidean distance and Manhattan distance, and the chosen distance metric depends on

the type of features in the dataset. It is important to indicate that the selected value of K, which represents the number of neighbors, is crucial in finding balance between overfitting and underfitting of KNN classifiers. If the chosen value of K is small, it produces KNN classifier that has low bias but very high variance, and if the chosen value of K is high, it produces KNN classifier that has high bias but low variance. One of the methods to determine the appropriate value of K is cross validation method where small portion of the training dataset is chosen as validation dataset which is used to evaluate the performance of the KNN model under different values of K and the value of K that produces the best performance on the validation dataset is selected.

After determination of appropriate value of K, voting scheme is used to determine the class of a new instance and the class is determined by the majority vote among the K nearest neighbors. The level of performance of KNN model can be improved by scaling of the features in the dataset, selection of odd value for K, and application of distance-weight during voting among the k nearest neighbors. The KNN algorithm is summarized in the following steps:

1. Choose a distance metric and use it to identify the K nearest neighbors out of N training vectors, irrespective of the class label.
2. Out of these K cases, identify the number of vectors, $K_i$, that belongs to class $C_i$, where, i = 1, 2..., C
3. Assign x to class $C_i$, with the maximum number of nearest neighbors.

In this project, four different types of KNN classifiers were created which are a classifier with all features in the data, classifiers with features selected from information gain, random forest and lasso regression feature selection techniques. The classifiers were built through 'caret' package in R, while total sample sizes were 44667, out of which 70% were selected as training set and remaining 30% as test set. The validation dataset is selected from the training set, and the performances of the classifiers are evaluated and compared. Table 4.1 show the performances of the different KNN models with their evaluation metrics;

| | All features (171) | I.G features (94) | R.F features (30) | Lasso Reg. features (68) |
|---|---|---|---|---|
| Accuracy | 0.9948 | 0.9957 | 0.9967 | 0.9963 |
| Precision | 0.8076 | 0.8333 | 0.9014 | 0.8714 |
| Recall | 0.4158 | 0.5445 | 0.6336 | 0.6039 |
| F1 score | 0.5489 | 0.6586 | 0.7441 | 0.7133 |
| AUCROC | 0.7075 | 0.7718 | 0.8165 | 0.8016 |

Table 4.1: KNN models

Table 4.1 shows that the model constructed with features from random forest feature selection technique had the highest values in all the evaluation metrics, and in term of precision, KNN models demonstrated high level of performance compared to models from logistic regression and Naïve Bayes models. Figures 22-29 represent the ROC curves, and the plot of the accuracy level against the number of the nearest neighbors for each of KNN models.

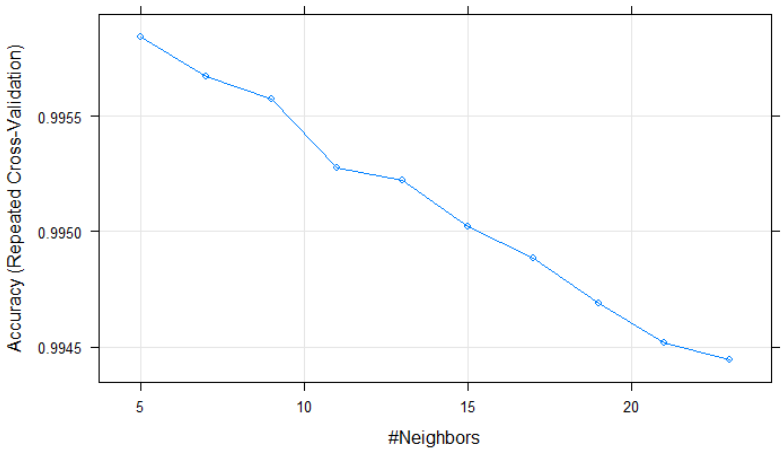


Figure 22: Accuracy against Neighbors – KNN (All features)

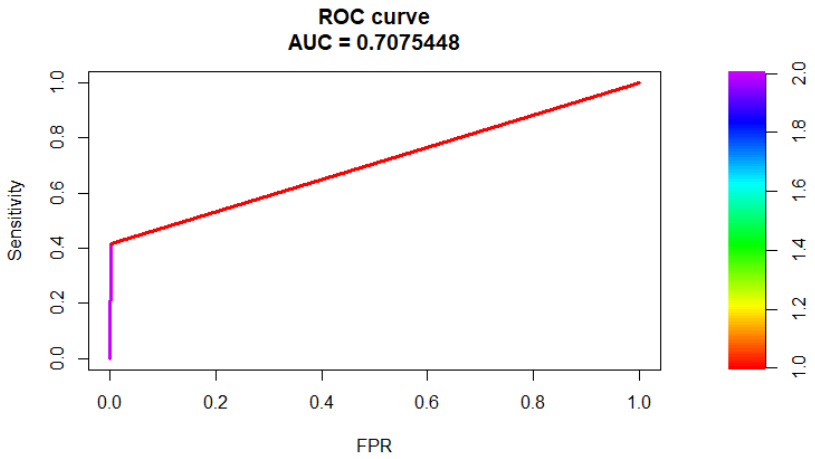


Figure 23: ROC curve – KNN (All features)

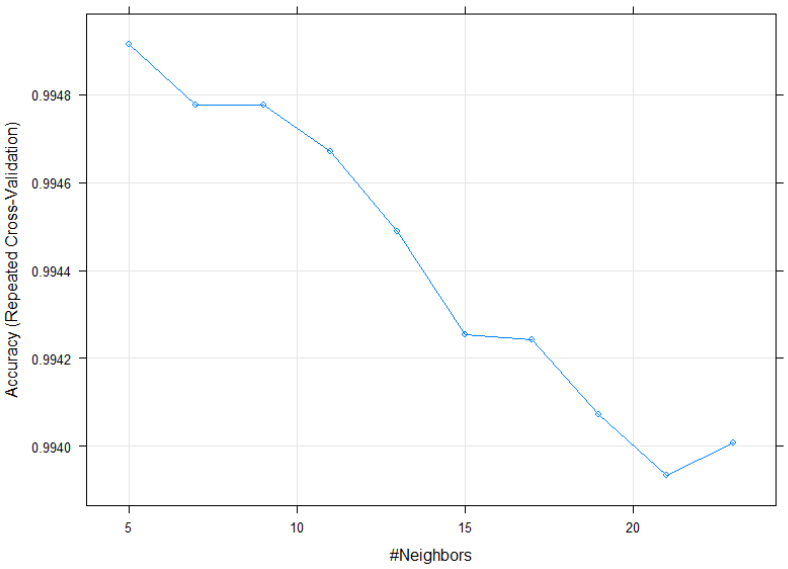


Figure 24: Accuracy against Neighbors – KNN (I.G features)

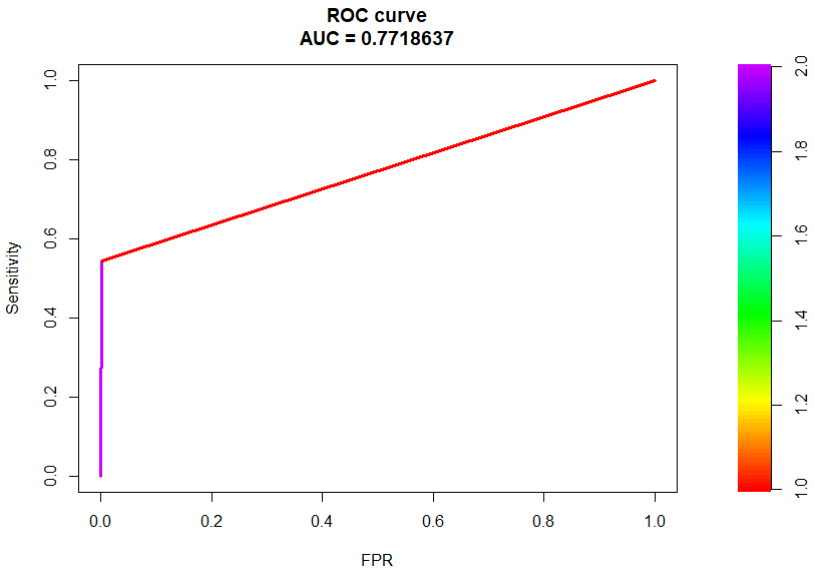


Figure 25: ROC curve – KNN (I.G features)

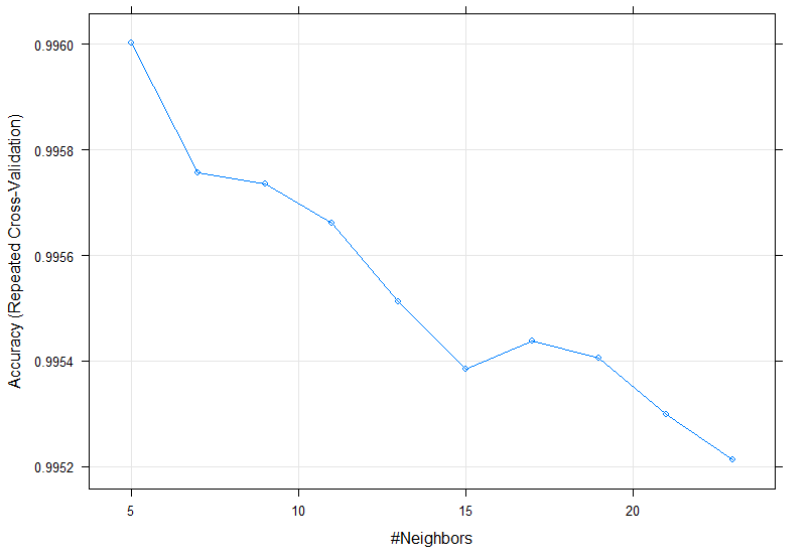


Figure 26: Accuracy against Neighbors – KNN (R.F features)

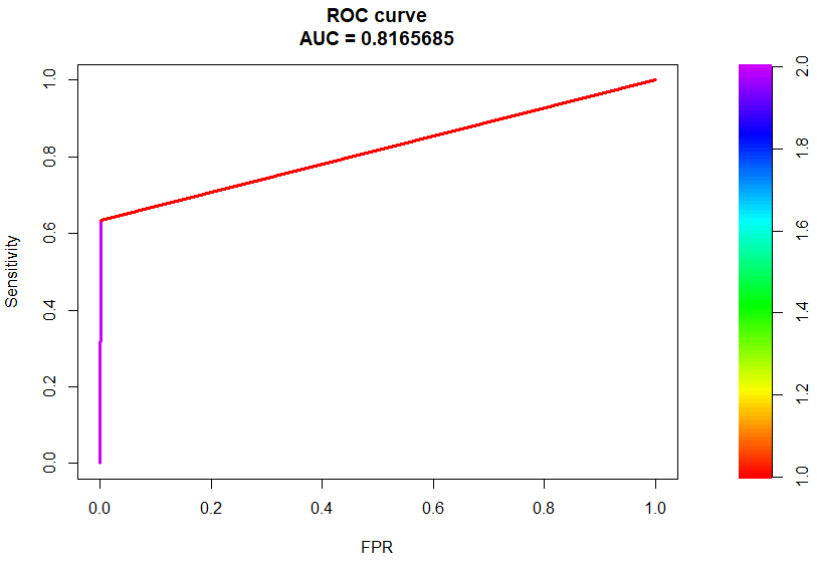


Figure 27: ROC curve – KNN (R.F features)

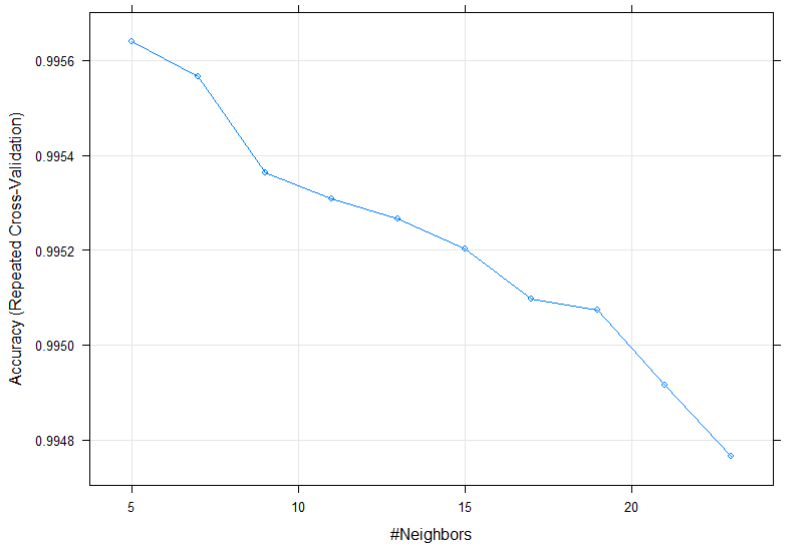


Figure 28: Accuracy against Neighbors – KNN (L.R features)

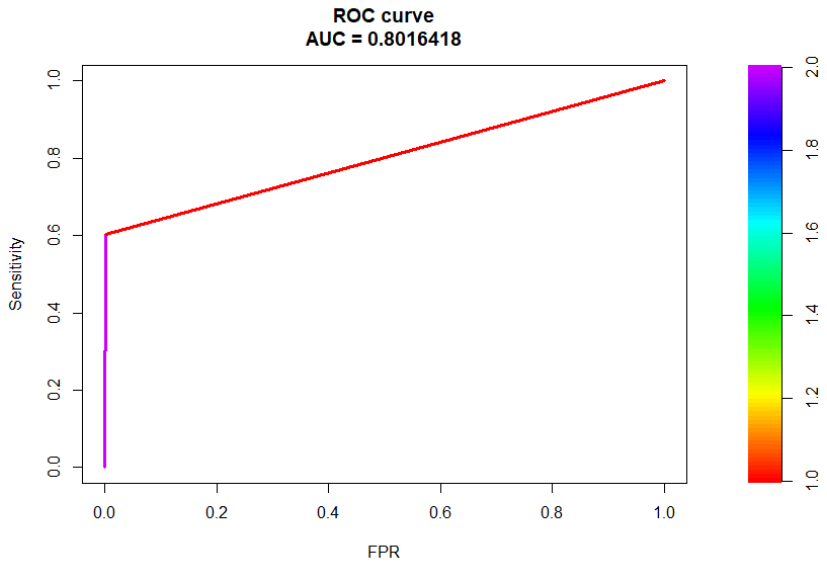


Figure 29: ROC curve – KNN (L.R features)

## 4.2 Support Vectors Machine classifier

Support Vectors Machine (which is abbreviated as SVM) can be used for classification task. It creates different hyperplanes that separate the data samples and amongst these different hyperplanes, it locates optimal hyperplane with maximum margin between the data samples that can accurately distinguish one class from the other class depending where the data sample is positioned on the side of the hyperplane [9].

The margin of a hyperplane can be defined as the distance between the separating hyperplane and the closest data points in the training samples that are closest to the hyperplane. The training data samples that guide and closest to the hyperplane are referred to as support vectors, and they determine the position and the orientation of the hyperplane. Figure 30 represents the concept of SVM in two-dimensional space, with linear separating hyperplane which separates the two classes with maximum margin.

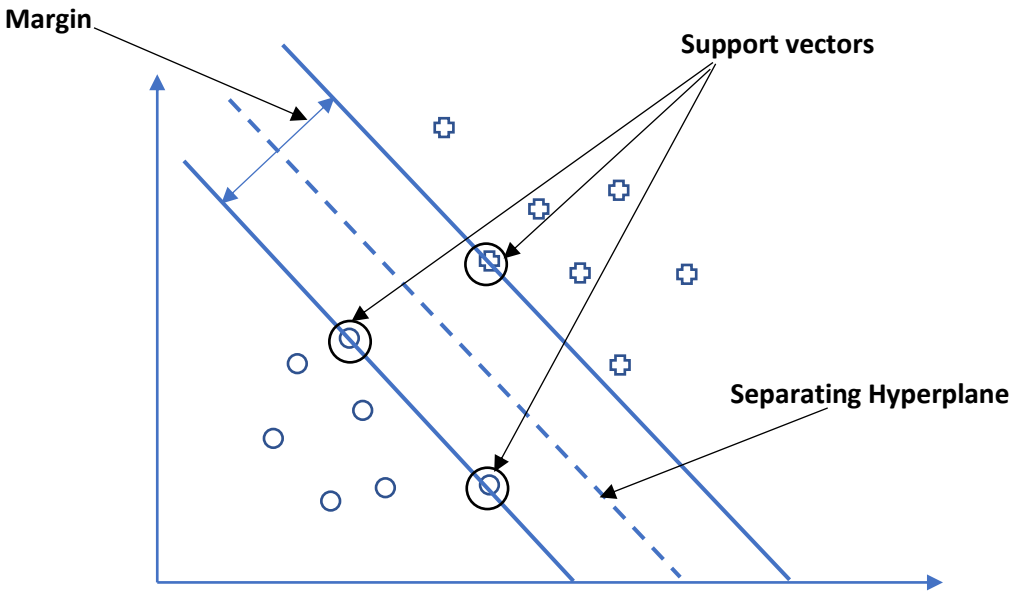


Figure 30: SVM with maximum margin

There are basically two different categories of linearly separable SVM, which are Hard-SVM and Soft-SVM. In Hard-SVM, there is a strong assumption that the data points in the training set are linearly separable, while in Soft-SVM, the linear constraints of Hard-SVM are relaxed and this makes soft-SVM suitable for partially linearly separable training set to enable the convergence of optimization in the existence of misclassification using appropriate cost penalization. The soft-SVM can be achieved through introduction of slack variable $\xi$ to the linear constraints [11].

Linear constraints (Hard-SVM):

$$w_0 + w^T x^{(i)} \geq 1 \; if \; y^{(i)} = 1 \tag{9}$$

$$w_0 + w^T x^{(i)} \leq -1 \; if \; y^{(i)} = -1 \tag{10}$$

$$for \; i = 1 \ldots N, where \; N \; is \; the \; number \; of \; samples \; in \; the \; dataset.$$

The simple interpretation of the two equations above is that, all positive samples should fall behind the positive hyperplane while, all the negative samples should fall behind the negative hyperplane.

Linear constraints with slack variable (Soft-SVM):

$$w_0 + w^T x^{(i)} \geq 1 - \xi^{(i)} \; if \; y^{(i)} = 1 \tag{11}$$

$$w_0 + w^T x^{(i)} \leq -1 + \xi^{(i)} \ if \ y^{(i)} = -1 \qquad (12)$$

$$for \ i = 1 \ldots N, where \ N \ is \ the \ number \ of \ samples \ in \ the \ dataset.$$

The new objective of maximization of margin with introduction of slack variable $\xi$ leads to generation of variable C which is used as margin control parameter and this can be represented with formula below:

$$\frac{1}{2}||w||^2 + C\left(\sum_i \xi^{(i)}\right) \qquad (13)$$

The quality of classification prediction and the penalty for misclassification errors can be controlled with the variable C. The large value of variable C represents large penalties for the misclassification errors while, small value of variable C represents small penalties for the misclassification errors and hence, variable C can be used to control the distance between separating hyperplanes and control the bias-variance trade-off as represented in the figure below;

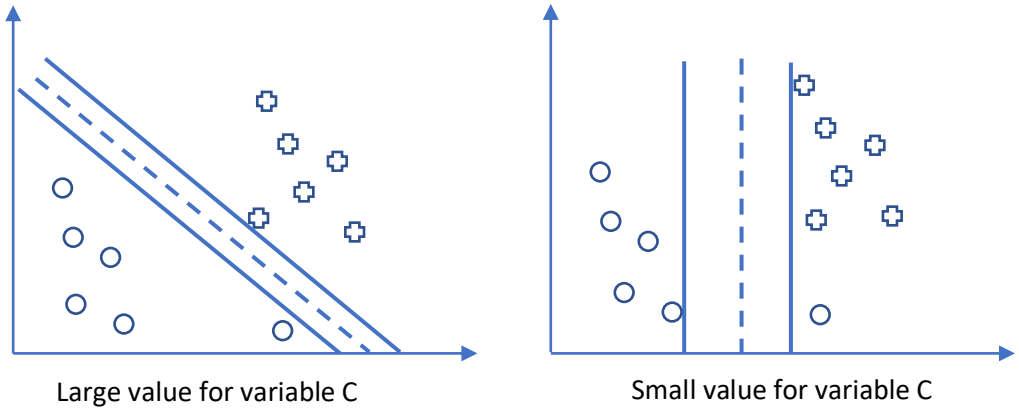


Figure 31: Effect of control parameter on SVM margin

### 4.2.1 Kernel SVM

The discussion on SVM before now has focused on the situation where the data samples is partially or fully linearly separable. Kernel SVM is a type of support vector machine that uses a linear classifier to classify data samples which is nonlinearly separable. The working principle of kernel is transformation of features of nonlinearly separable data samples into features that produces linearly separable data samples, and these

transformations are called kernels. Some of the most commonly used kernels are polynomial kernel, gaussian kernel, radial basis function, and sigmoid kernel.

In this part of the project, linear SVM classifiers are constructed with the assumption that the dataset is fully or partially linearly separable, and four different types of SVM classifiers were created which are a classifier with all features in the data, classifiers with features selected from information gain, random forest and lasso regression feature selection techniques. The classifiers were built through 'caret' package in R, while total sample sizes were 44667, out of which 70% were selected as training set and remaining 30% as test set. The validation dataset is selected from the training set, and the performances of the classifiers are evaluated and compared. Table 4.2 shows the performances of different linear SVM models with their evaluation metrics.

| | All features (171) | I.G features (94) | R.F features (30) | Lasso Reg. features (68) |
|---|---|---|---|---|
| Accuracy | 0.9962 | 0.9959 | 0.9959 | 0.9965 |
| Precision | 0.8493 | 0.8219 | 0.8405 | 0.9130 |
| Recall | 0.6138 | 0.5940 | 0.5742 | 0.6237 |
| F1 score | 0.7125 | 0.6896 | 0.6822 | 0.7411 |
| AUCROC | 0.8065 | 0.7965 | 0.7867 | 0.8115 |

Table 4.2: Linear SVM models

Table 4.2 shows that the model constructed with features from lasso regression feature selection technique had the highest values in terms of evaluation metrics compared to models constructed from other features. Figures 32-39 represents the ROC curves, and the plots of accuracy against cost for each of SVM models where cost represents a control parameter for misclassification error.

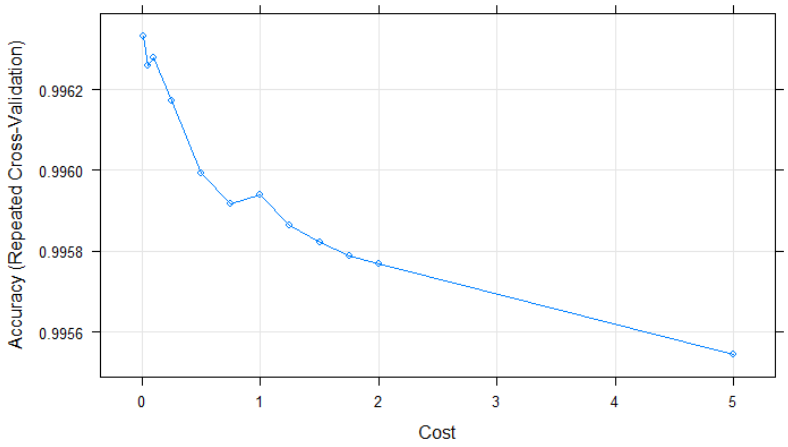


Figure 32: Accuracy against Cost – SVM (All features)

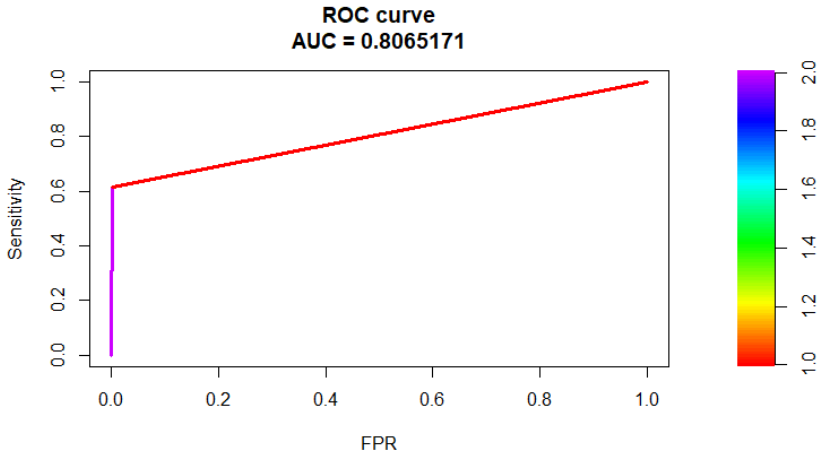


Figure 33: ROC curve – SVM (All features)

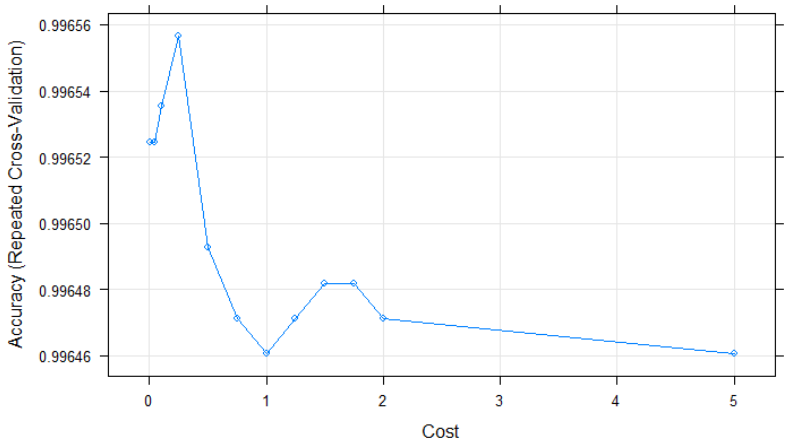


Figure 34: Accuracy against Cost – SVM (I.G features)

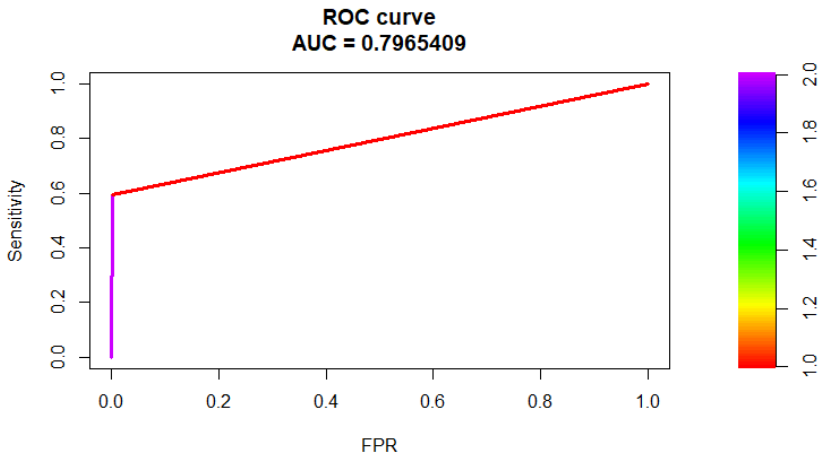


Figure 35: ROC curve – SVM (I.G features)

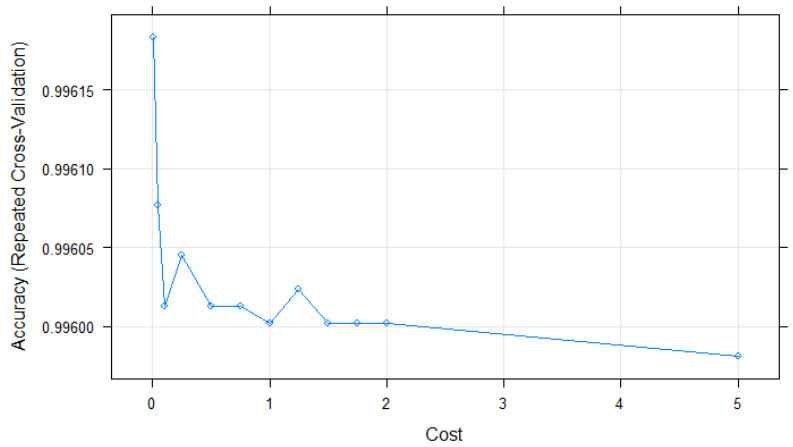


Figure 36: Accuracy against Cost – SVM (R.F features)

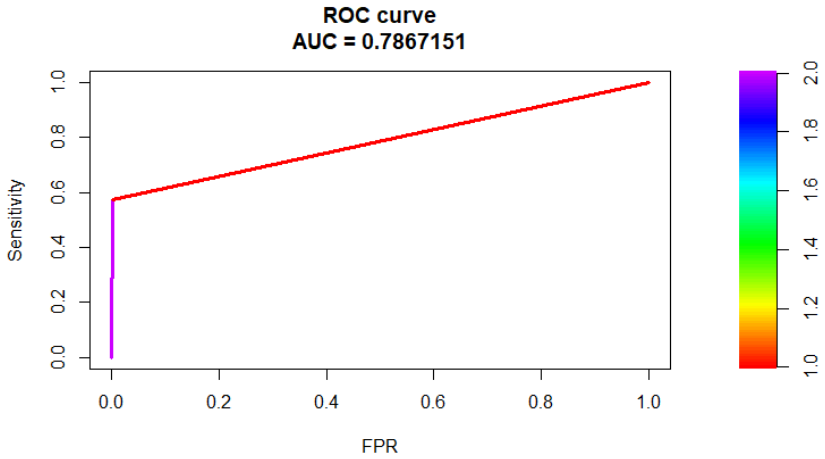


Figure 37: ROC curve – SVM (R.F features)

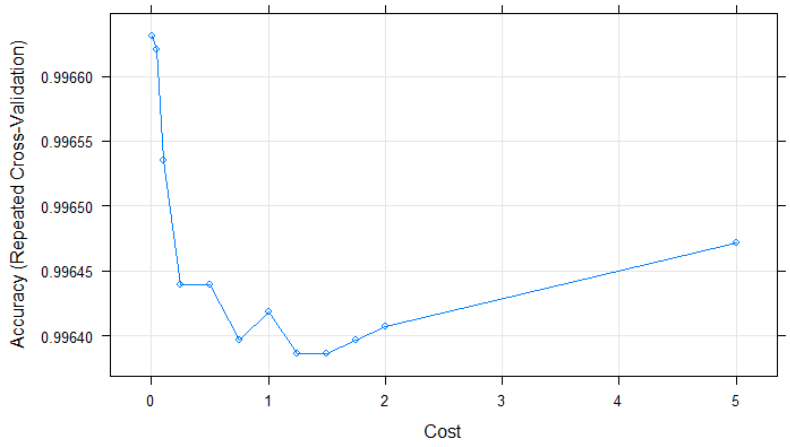


Figure 38: Accuracy against Cost – SVM (L.R features)

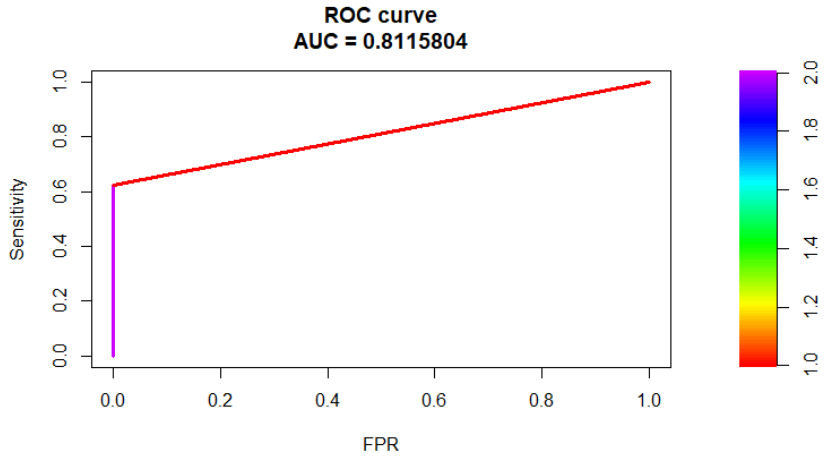


Figure 39: ROC curve – SVM (L.R features)

## 4.3 Ensemble Learning

The main objective of the ensemble learning is the combination of different classifiers to produce a single classifier with better performance evaluation metrics than individual classifier alone. Ensemble method produces a classifier with reduction in variance, bias and improved predictive power. There are various techniques used by ensemble method in achieving its objective and the most common three of these techniques are Bagging, Boosting and Stacking techniques.

### 4.3.1 Bagging

This is one of the techniques of ensemble learning which involves building N number of classifiers. The training samples used for each classification model is a subset of the initial training set and each subset is drawn at random with replacement from the initial training set, because of this the bagging technique is also called bootstrap aggregating. The results of predictions of each N numbers of classifiers are used to provide the result of the final prediction depending on the nature of the task. For a regression task, the final prediction is the average of predictions from N classifiers, while for a classification task, the final prediction is achieved through majority voting scheme of the predictions from N classifiers [9].

A random forest classifier is an example of ensemble method that makes use of bagging technique, where a decision tree represents the individual classifiers that are built with a random subset of initial training set, and in addition, each decision tree is built using random features subsets of data features. Bagging technique is effective in reducing variance of a classifier, but not effective in reducing bias of a classifier. The concept of bagging is illustrated in the Figure 40:

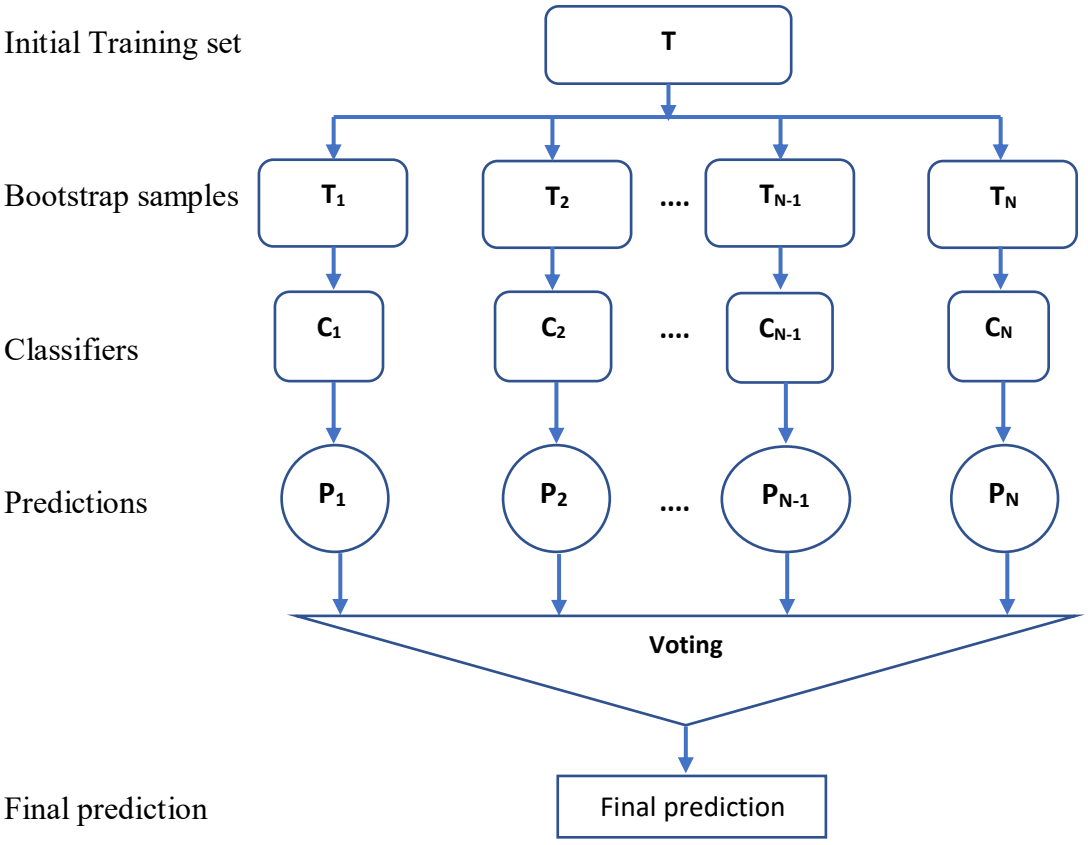


Figure 40: The Concept of Boosting technique of Ensemble Learning

### 4.3.2 Boosting

The main objective of Boosting technique of ensemble learning is to convert weak learners to strong learners by focusing on training of samples that are difficult to classify, and this is achieved by giving more weight to samples that were previously misclassified and reducing the weight of correctly classified samples. Weak learners, such as decision trees, are the learners that have slightly better performance evaluation metrics than random guessing. In the same way as bagging, the training samples for boosting is a subset of the initial training set. However, in contrast to bagging, each subset in boosting is drawn at random without replacement from the initial training set. Boosting technique can be an effective method of reducing bias of a model and a typical boosting technique is summarized with the following steps [9]:

1. A random subset of training sample t1 is drawn without replacement from the initial training set T, and the training sample t1 is used to train weak learner L1.
2. The second subset of training sample t2 is drawn without replacement from the initial training set T, and half of the previously misclassified samples is added to the drawn training samples t2 to train a weak learner L2.
3. The misclassified samples from L1 and L2 are used to form training sample t3 which is used to train another weak leaner T3.
4. The final prediction is decided based on the majority vote form the weak learners L1, L2, and L3.

### 4.3.3 Stacking

This is one of the techniques of ensemble learning which involves using the predictions of other multiple learning algorithms as the input features for training the main learning algorithm. The initial complete dataset is used for training multiple learning algorithms and this technique is different from boosting and bagging techniques in the sense that it does not involve drawing random subset of samples from initial complete dataset for training base multiple learning algorithms and it does not involve using majority voting scheme in final prediction. Stacking can be effective in reducing bias and variance simultaneously, depending on the different learning algorithms used at base level and practically logistic regression is often used as combiner learning algorithm. The concept of stacking is illustrated in Figure 41:

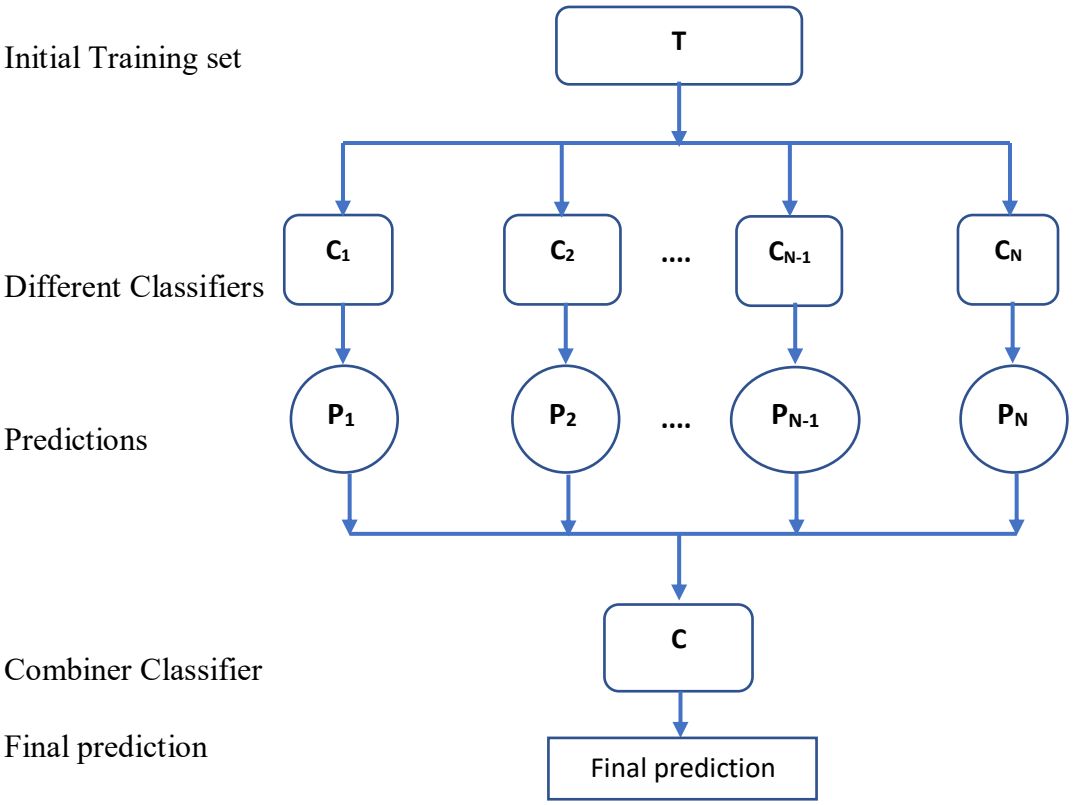


Figure 41: The Concept of Stacking technique of Ensemble Learning

It is important to know that stacking produces biggest gains when the base learners that are being stacked have high variability and when the predicted values of the base learners are uncorrelated.

The dataset of this project was used to execute the two techniques of ensemble learning that were previously discussed, which are bagging and boosting. Four different bagging classifiers were created which are a classifier with all features in the data, classifiers with features selected from information gain, random forest and lasso regression feature selection techniques. The classifiers were built through 'ipred' package in R, while total sample sizes were 44667, out of which 70% were selected as training set and remaining 30% as test set. The performances of the classifiers are evaluated and compared. Table 4.3 shows the performances of the different bagging models with their evaluation metrics.

| | All features (171) | I.G features (94) | R.F features (30) | Lasso Reg. features (68) |
|---|---|---|---|---|
| Accuracy | 0.9970 | 0.9974 | 0.9969 | 0.9969 |
| Precision | 0.8604 | 0.8764 | 0.8488 | 0.8409 |
| Recall | 0.7326 | 0.7722 | 0.7227 | 0.7326 |
| F1 score | 0.7913 | 0.8210 | 0.7806 | 0.7830 |
| AUCROC | 0.8658 | 0.8857 | 0.8608 | 0.8658 |

Table 4.3: Bagging models

Table 4.3 shows that the model constructed with features from information gain feature selection technique had the highest values in terms of evaluation metrics. Figures 42-45 represent the ROC curves of Bagging models:

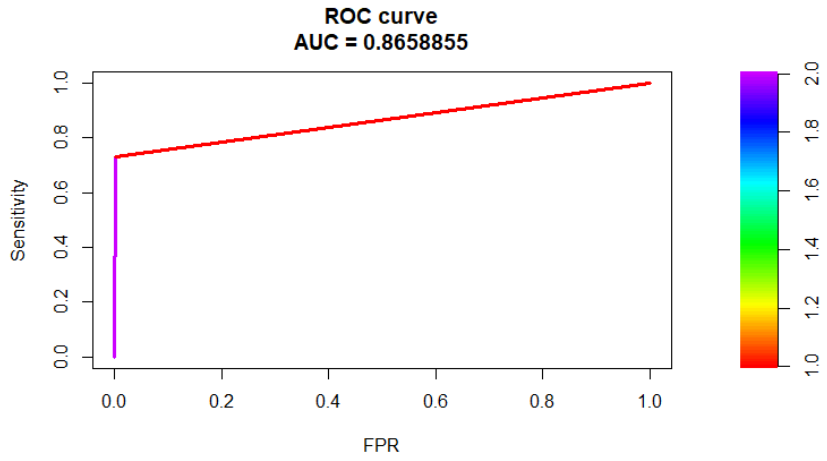


Figure 42: ROC curve – Bagging (All features)

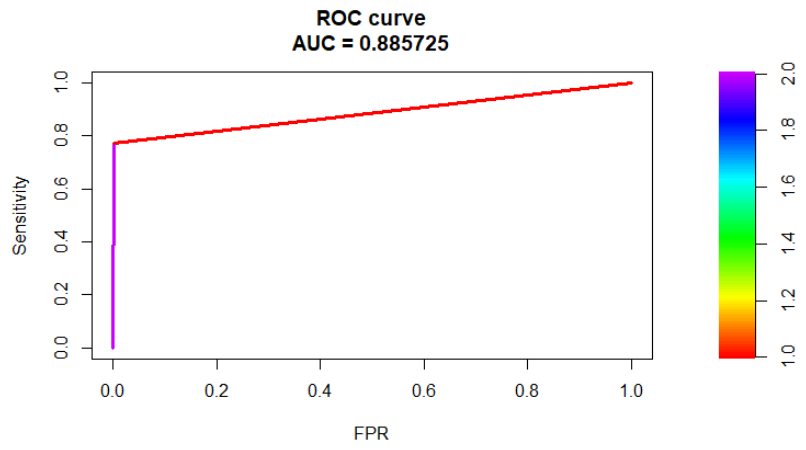

Figure 43: ROC curve – Bagging (I.G features)

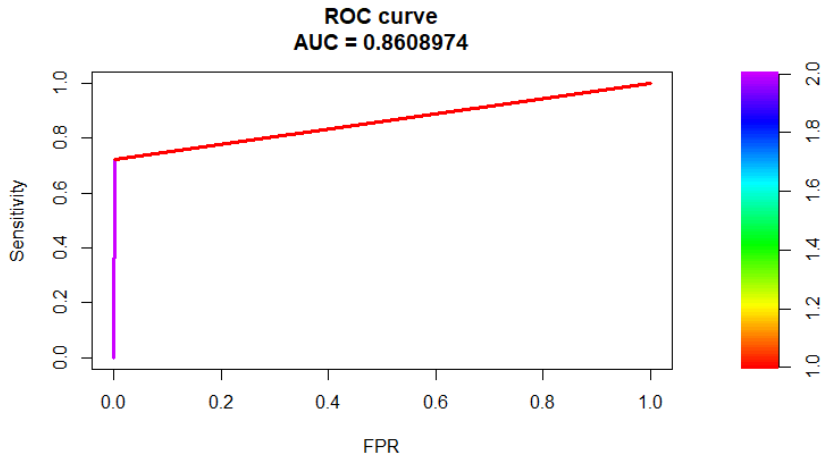


Figure 44: ROC curve – Bagging (R.F features)

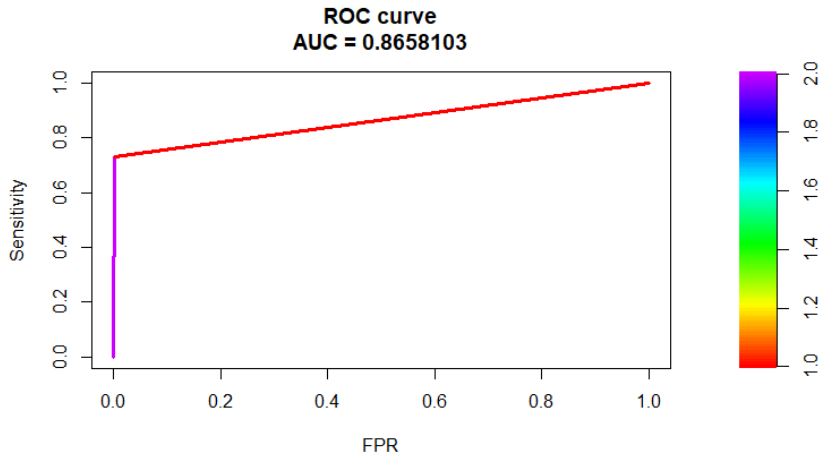


Figure 45: ROC curve – Bagging (L.R features)

The dataset of this project was used to build gradient boosting models and four different boosting classifiers were created which are a classifier with all features in the data, classifiers with features selected from information gain, random forest and lasso regression feature selection techniques. The classifiers were built through 'gbm' package in R, while total sample sizes were 44667, out of which 70% were selected as training set and remaining 30% as test set. The performances of the classifiers are

evaluated and compared. Table 4.4 shows the performances of the different boosting models with their evaluation metrics.

| | All features (171) | I.G features (94) | R.F features (30) | Lasso Reg. features (68) |
|---|---|---|---|---|
| Accuracy | 0.9957 | 0.9956 | 0.9957 | 0.9954 |
| Precision | 0.8437 | 0.8115 | 0.8666 | 0.8030 |
| Recall | 0.5346 | 0.5544 | 0.5148 | 0.5247 |
| F1 score | 0.6544 | 0.6587 | 0.6459 | 0.6346 |
| AUCROC | 0.9712 | 0.9709 | 0.9715 | 0.9802 |

Table 4.4: Boosting models

Table 4.4 shows that there is no strong difference in the evaluation metrics of boosting models constructed with all the features and with the features from the feature selection techniques. The boosting models had high evaluation metrics in terms of accuracy, precision and AUCROC. However, the recall and F1 score values are low compare to the recall and F1 score values in bagging models which is also an ensemble learning technique, and this is an evidence that boosting method builds low bias models while bagging method builds low variance models. Figures 46-49 represent the ROC curves of the Boosting models.

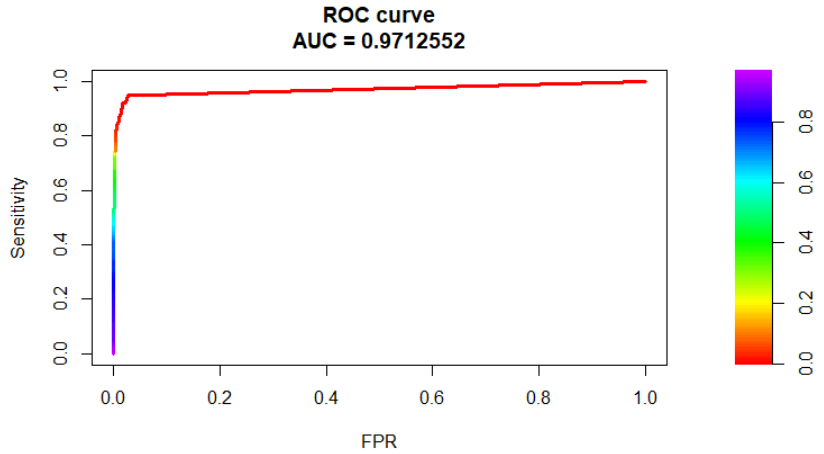


Figure 46: ROC curve – Boosting (All features)

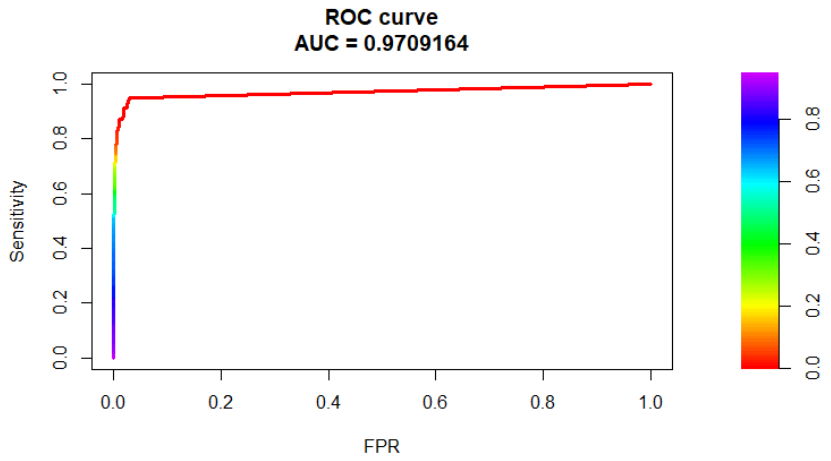


Figure 47: ROC curve – Boosting (I.G features)

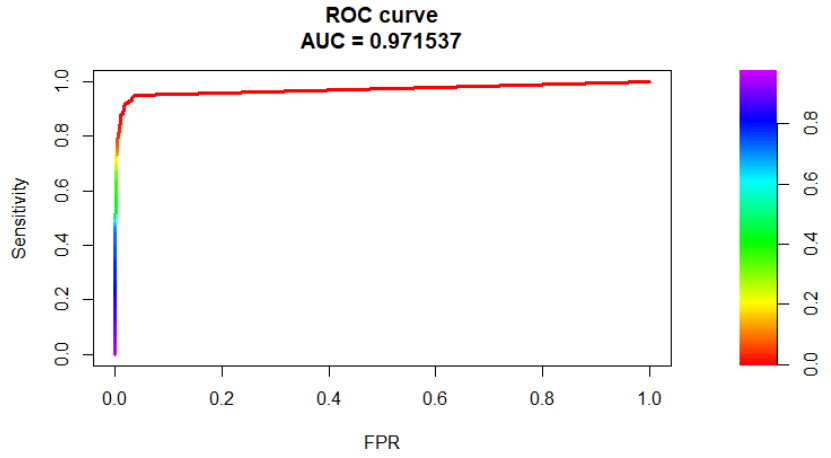


Figure 48: ROC curve – Boosting (R.F features)

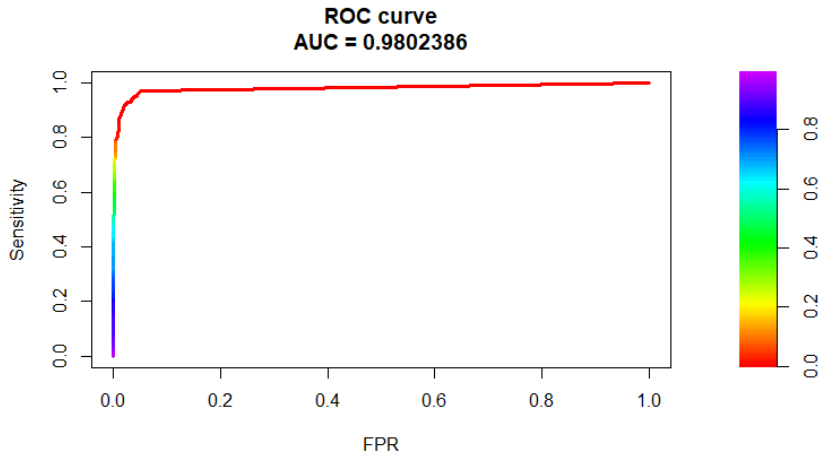


Figure 49: ROC curve – Boosting (L.R features)

# Chapter five

## 5.1 Summary and Conclusion:

This research work makes use of a publicly available dataset to examine the behavior of different machine learning methods to classify a new data sample into the positive or negative class, in the area of contributing and improving the activities of predictive maintenance in order to optimize the business advantage of the predictive maintenance. The focus of this research work is to achieve minimum type I and type II errors through selection of important features and an effective machine learning method. Maximizing the benefit of machine learning does not only involve selecting the appropriate machine learning method but also involves the selection of appropriate and important features in the dataset, most especially when there is a high dimensionality in the dataset.

In this thesis work, three most important steps were carried out on the dataset before feeding the dataset into different machine learning methods. The first step was the exploratory analysis of the dataset which involved examining the distribution of the dependent variable, and discovery of the missing data. The second step involved methods of handling missing data, where two popular methods of handling missing data were used, and these are usage of Missing at Random (MAR) mechanism for inputting missing values and deletion of cases with missing value. The third step involved feature engineering where feature selection techniques such as information gain, random forest and lasso regression were used to select the important features from the dataset.

This research work involved examination of six different machine learning methods for the classification task, and these methods were logistic regression, Naïve Bayes classifier, KNN, Linear SVM, Bagging and Boosting methods of Ensemble learning. Four different types of models were created for each method, models with all the features in the data, models with features selected from information gain, random forest and lasso regression. The results of these methods were compared through five major performance evaluation metrics which were Accuracy, Precision, Recall, F1 score and Area Under ROC curve. Within method comparison of models, and between methods comparison of models were carried out.

In terms of accuracy, all the considered models performed very well, but the bagging method of ensemble learning had the highest performance in term of accuracy with the accuracy of 99.74% and followed by KNN model with the accuracy of 99.67%. In terms of precision and recall which are the focus of this research work, Linear SVM model had the highest performance with the precision value of 91.30% and followed by KNN model with the precision value of 90.14%. The Naïve Bayes model had the highest performance in terms of recall with the recall value of 90.10% and followed by Bagging model with the recall value of 77.22%. However, the precision of Naïve Bayes models was low because of relatively great false positive number compared to true positive number and the model did poorly in terms of precision and F1 score as the evaluation metrics, thereby making the model the least performing model out of the six models that were considered.

In terms of AUCROC, all the considered models performed relatively well, but the boosting method of ensemble learning had the highest performance in term of AUCROC with the AUCROC value of 0.9802 and followed by Logistics regression model with the AUCROC value of 0.9659. The F1 score evaluation metric is the harmonic mean of both precision and recall and it was used to select the best performing model out of the six models because the focus of this research work is to achieve minimum type I and type II errors where high precision and recall are contributing factor respectively. The Bagging method had the highest performance in term of F1 score with the F1 score of 82.10% and followed by KNN models.

The results of this study demonstrated the importance of feature engineering in improving the performance of the machine learning models, and the results also suggested that Ensemble learning methods are efficient in reducing variance and bias in the dataset thereby producing effective predictive models that reduces type I and type II errors.

This work can be improved in the future by investigating the behavior of the other latest machine learning technique and most especially the artificial neural networks techniques. Also, this work can be improved by investigating the behavior of machine learning methods using complete dataset without missing values or with minimum percentage of missing data, and the improvement can be extended to other methods of handling missing data where there is an incomplete dataset.

## References:


[1] Wikipedia; Maintenance Technical.

"https://en.wikipedia.org/wiki/Maintenance_(technical)", [Online; accessed 15-May-2019].

[2] British Standard Institution, "Glossary of Terms Used in Terotechnology.", BS 3811, United Kingdom. (1993).

[3] Finnish Standardization Association, "Maintenance. Maintenance terminology, 2nd edition.", SFS-EN 13306, Helsinki, Finland. (2010).

[4] MARCUS BENGTSSON, ERIK OLSSON, PETER FUNK, and MATS JACKSON, "Technical Design of Condition Based Maintenance System: A Case Study Using Sound Analysis and Case-Based Reasoning.", Maintenance and Reliability Conference, Knoxville, USA, (2004).

[5] British Standards Institution, "Maintenance - Maintenance terminology.", BS-EN-13306, United Kingdom. (2010).

[6] JANTUNEN, E., ARNAIZ, A., BAGLEE, D. and FUMAGALLI, L., "Identification of wear statistics to determine the need for a new approach to maintenance.", Euro Maintenance Conference, Helsinki, Finland, (2014).

[7] KHAIRY A.H. KOBBACY and D.N. PRABHAKAR MURTHY, "Complex System Maintenance Handbook.", Springer-Verlag London Limited, (2008).

[8] TOM M. MITCHELL, "Machine Learning.", McGraw-Hill Science/Engineering/Math, (1997).

[9] SEBASTIAN RASCHKA, and VAHID MIRJALILI, "Python Machine Learning.", Packt Publishing Ltd., (2017).

[10] JUDITH HURWITZ, and DANIEL KIRSCH, "Machine Learning for dummies.", John Wiley & Sons, Inc., (2018).

[11] SHAI SHALEV-SHWARTZ AND SHAI BEN-DAVID, "Understanding Machine Learning: From Theory to Algorithms.", Cambridge University Press., (2014).

[12] MOHAMMED A. NOMAN, EMAD S. ABOUEL NASR, ADEL AL-SHAYEA, and HUSAM KAID, "Overview of predictive condition-based maintenance using bibliometric indicators.", Journal of King Saud University – Engineering Sciences 31, (2019): p.356-367.

[13] MARZIO, ENRICO, and LUCA, "CBM optimization by means of genetics algorithms and MC simulation.", Journal of Reliability Engineering & System Safety, Vol. 77 Issue 2, (2002): p.151-165.

[14] S.K. Yang, "An experiment of state estimation of predictive maintenance using Kalman filter on a DC motor.", Journal of Reliability Engineering &System Safety, Vol. 75 Issue 1, (2002): p.103-111.

[15] DHEERAJ BANSAL, DAVID J. EVANS, and BARRIE JONES, "A real-time predictive maintenance system for machine systems.", International Journal of Machine Tools and Manufacture, Vol. 44 Issues 7-8, (2004): p.759-766.

[16] WENBIN WANG, "A two-stage prognosis model in condition-based maintenance.", European Journal of Operational Research, Vol. 182 Issue 3, (2007): p.1177-1187.

[17] STEFANO, ROBERTO, and SERGIO, "Application of neural networks to condition based maintenance: a case study in the textile industry.", 8th IFAC Workshop on Intelligent Manufacturing Systems, Vol. 40 Issue 3, (2007): p.147-152.

[18] W.WANG, and W.ZHANG, "An asset residual life prediction model based on expert judgments.", European Journal of Operational Research, Vol. 188 Issue 2, (2008): p.496-505.

[19] Y. G. LI, "Gas turbine performance prognostic for condition-based maintenance.", Journal of Applied Energy, Vol. 86 Issue 10, (2009): p.2151-2161.

[20] ABD KADIR, SHARIFAH, and TAKASHI, "Predicting remaining useful life of rotating machinery based artificial neural network.", Journal of Computers & Mathematics with Applications, Vol. 60 Issue 4, (2010): p.1078-1087.

[21] YING PENG, and MING DONG, “A prognosis method using age-dependent hidden semi-Markov model for equipment health prediction.”, Journal of Mechanical Systems and Signal Processing, Vol. 25 Issue 1, (2011): p.237-252.

[22] ACHMAD WIDODO, and BO-SUK YANG, “Machine health prognostics using survival probability and support vector machine.”, Journal of Expert Systems with Applications, Vol. 38 Issue 7, (2011): p.8430-8437.

[23] JINQUI HU, LAIBIN ZHANG, and WEI LIANG, “Opportunistic predictive maintenance for complex multi-component systems based on DBN-HAZOP model.”, Journal of Process Safety and Environmental Protection, Vol. 90 Issue 5, (2012): p.376-388.

[24] HACK-EUN KIM, ANDY C. C. TAN, JOSEPH MATHEW, and BYEONG-KEUN CHOI, “Bearing fault prognosis based on health state probability estimation.”, Journal of Expert Systems with Applications, Vol. 39 Issue 5, (2012): p.5200-5213.

[25] JIN YUAN, and XUEMEI LIU, “Semi-supervised learning and condition fusion for fault diagnosis.”, Journal of Mechanical Systems and Signal Processing, Vol. 38 Issue 2, (2013): p615-627.

[26] T. PRAVEENKUMAR, M. SAIMURUGAN, P. KRISHNAKUMAR, and K. I. RAMACHANDRAN, “Fault Diagnosis of Automobile Gearbox Based on Machine Learning Techniques.”, Journal of Procedia Engineering, Vol. 97, (2014): p.2092-2098.

[27] MARTHA A. ZAIDAN, ROBERT F. HARRISON, ANDREW R. MILLS, and PETER J. FLEMING, “Bayesian Hierarchical Models for aerospace gas turbine engine prognostics.”, Journal of Expert Systems with Applications, Vol. 42 Issue 1, (2015): p.539-553.

[28] HUI, and JIANCHAO, “Real-time prediction of remaining useful life and preventive opportunistic maintenance strategy for multi-component systems considering stochastic dependence.”, Journal of Computers & Industrial Engineering, Vol. 93, (2016): p.192-204.

[29] RICCARDO, RICCARDO, PIETRO, MARCO, and SIMONE, "Data Mining and Machine Learning for Condition-based Maintenance.", Journal of Procedia Manufacturing, Vol. 11, (2017): p.1153-1161.

[30] CHRISTOPHER GONDEK, DANIEL HAFNER, and OLIVER R. SAMPSON, "Prediction of Failures in the Air Pressure System of Scania Trucks using Random Forest and Feature Engineering.", Conference: The 15th International Symposium on IDA, (2016).

[31] MICHELE ALBANO, ERKKI JANTUNEN, GREGOR PAPA, and URKO ZURUTUZA, "The Mantis Book: Cyber Physical System Based Proactive Collaborative Maintenance.", River Publishers., (2019).

[32] Medium; Dealing with Missing Data using R.

"https://medium.com/coinmonks/dealing-with-missing-data-using-r-3ae428da2d17", [Online; accessed 10-June-2019].

[33] TowardsDataScience; Feature Selection Using Random Forest.
"https://towardsdatascience.com/feature-selection-using-random-forest-26d7b747597f", [Online; accessed 21-May-2019].

[34] Wikipedia; Mutual Information.

"https://en.wikipedia.org/wiki/Mutual_information", [Online; accessed 12-May-2019].

[35] Interpretable Machine Learning; Logistic regression.

"https://christophm.github.io/interpretable-ml-book/logistic.html", [Online; accessed 10-July-2019].

[36] Mice Vignettes

"https://www.gerkovink.com/miceVignettes/", [Online: accessed 10-June-2019].

[37] CRAIG K. ENDERS, "Applied Missing Data Analysis.", The Guilford Press., (2010).